Perceptual Video Restoration: Task-Specific versus Pre-trained Large Generative Models

by

Nasrin Rahimi

A Dissertation Submitted to the
Graduate School of Sciences and Engineering
in Partial Fulfillment of the Requirements for
the Degree of

Doctor of Philosophy

in

Electrical and Electronics Engineering

KOÇ ÜNİVERSİTESİ

September 1, 2025

Perceptual Video Restoration:
Task-Specific versus Pre-trained Large
Generative Models

Koç University
Graduate School of Sciences and Engineering
This is to certify that I have examined this copy of a doctoral dissertation by

Nasrin Rahimi

and have found that it is complete and satisfactory in all respects,
and that any and all revisions required by the final
examining committee have been made.

Committee Members:

Prof. A. Murat Tekalp (Advisor)

Prof. Yücel Yemez

Assist. Prof. Zafer Doğan

Prof. Hasan Fehmi Ateş

Prof. Gözde Bozdağı Akar

Date: _____

To my beloved family and all who stood by me with their support, patience, and love throughout this long journey.

ABSTRACT

Perceptual Video Restoration:
Task-Specific versus Pre-trained Large
Generative Models
Nasrin Rahimi
Doctor of Philosophy in Electrical and Electronics Engineering
September 1, 2025

Video restoration problems are ill-posed inverse problems with a very large null space. Perceptual video restoration refers to choosing a feasible solution that is perceptually pleasing and temporally coherent. This thesis compares two paradigms for perceptual video restoration: supervised task-specific generative models and pre-trained large generative models applied in a zero-shot manner with some inference-time adaptation across diverse video restoration tasks.

In the first part, we present a novel task-specific Video Super-Resolution (VSR) model trained under a perception-aware framework, which trades fidelity for spatio-temporal perceptual quality. Our architecture employs two discriminators: one that focuses on the realism of spatial texture, and the other on the perceptual naturalness of motion. This design is supported by a hypothesis called perceptual straightening of natural videos, which states that natural image sequences that appear curved in the intensity domain tend to follow straighter trajectories in the perceptual domain of human visual system. Extensive experiments demonstrate that this model achieves superior spatio-temporal perceptual quality compared to baseline approaches.

In the second part, we explore a zero-shot, training-free paradigm to adapt large pre-trained image-based latent diffusion models (LDMs), for general-purpose video restoration tasks including video super-resolution, inpainting, deblurring, motion blur removal, and a combination of aforementioned tasks. While these models are rather flexible and offer quite good spatial generative capabilities, being inherently image-based, suffer from temporal inconsistencies when applied frame by frame, and mechanisms for enforcing temporal coherence remain relatively underexplored

in this zero-shot setting. To address this, we introduce three contributions: (1) a cross-frame attention approach to foster feature propagation without any motion estimation; (2) perceptual straightness regularization to guide sampling in finding perceptually smoother temporal paths, and, (3) a multi-path ensemble sampling scheme that improves overall quality—including temporal consistency—during the diffusion denoising process.

Quantitative and qualitative results reveal a trade-off between task-specific models and general zero-shot large-scale models: task-specific models achieve the best fidelity and perceptual scores when test degradations match training, but their performance drops under degradation kernel or dataset shift, where the zero-shot model remains competitive or superior and provides a flexible, training-free option—albeit with higher runtime. This thesis presents a detailed exploring of both paradigms, with a focus on the different approaches employed to address the challenge of temporal consistency—leveraging explicit training-based mechanisms in task-specific models and introducing inference-time temporal consistency techniques within the zero-shot diffusion framework.

ÖZETÇE

Doktora Tez Başlığı
Nasrin Rahimi
Elektrik ve Elektronik Mühendisliği, Doktora
1 Eylül 2025

Video restorasyonu problemleri, çok büyük bir null uzaya sahip, iyi tanımlanmamış ters problemlerdir. Algısal video restorasyonu, algısal olarak tatmin edici ve zamansal olarak tutarlı uygulanabilir bir çözümün seçilmesine karşılık gelir. Bu tez, algısal video restorasyonu için iki paradigmayı karşılaştırmaktadır: denetimli, göreve özgü üretici modeller ve önceden eğitilmiş büyük üretici modellerin çeşitli video restorasyon görevlerinde sıfır-atış (zero-shot) yaklaşımıyla, bazı çıkarım (inference) zamanı uyarlamalarıyla uygulanması.

İlk kısımda, algı farkındalıklı bir çerçevede eğitilen yeni bir göreve özgü Video Süper-Çözünürlük (VSR) modeli sunuyoruz. Bu yaklaşım, doğruluğu uzamsal-zamansal algısal kalite ile değiş tokuş etmektedir. Mimari tasarımımız, biri uzamsal dokuların gerçekçiliğine, diğeri ise hareketin algısal doğallığına odaklanan iki ayrıştırıcı (discriminator) kullanmaktadır. Bu tasarım, doğal görüntü dizilerinin yoğunluk uzayında eğri görünebilmesine rağmen insan görsel sisteminin algısal alanında daha düz yörüngeleri takip etme eğiliminde olduğunu öne süren "doğal videoların algısal düzleşmesi" hipoteziyle desteklenmektedir. Kapsamlı deneyler, bu modelin taban yöntemlere kıyasla üstün uzamsal-zamansal algısal kalite elde ettiğini göstermektedir.

İkinci kısımda, önceden eğitilmiş büyük ölçekli görüntü tabanlı latent difüzyon modellerini (LDM'ler), video süper-çözünürlük, inpainting, bulanıklık giderme, hareket bulanıklığı giderme ve bu görevlerin birleşimlerini içeren genel video restorasyonu için sıfır-atış, eğitim gerektirmeyen bir paradigma altında uyarlamayı araştırıyoruz. Bu modeller esnek olup iyi uzamsal üretim yetenekleri sunmasına rağmen, doğaları gereği görüntü tabanlı olduklarından kare-kare uygulandığında zamansal tutarsızlıklardan muzdarip olmaktadır. Ayrıca, zamansal tutarlılığı sağlamaya yönelik mekanizmalar bu sıfır-atış bağlamında görece az incelenmiştir. Bu sorunu ele almak için üç katkı

sunuyoruz: (1) herhangi bir hareket kestirimi olmadan özellik yayılımını kolaylaştıran çapraz-kare dikkat (cross-frame attention) yaklaşımı; (2) örnekleme sürecinde algısal olarak daha düzgün zamansal yolları bulmaya rehberlik eden algısal düzleşme düzenlemesi; (3) difüzyon denoising süreci boyunca genel kaliteyi—zamansal tutarlılık dahil—artıran çok-yollu (multi-path) topluluk (ensemble) örnekleme şeması.

Nicel ve nitel sonuçlar, göreve özgü modeller ile genel sıfır-atış büyük ölçekli modeller arasında bir ödünleşim (trade-off) olduğunu ortaya koymaktadır: göreve özgü modeller, test bozulmaları eğitimle eşleştiğinde en iyi doğruluk ve algısal skorları elde eder, ancak bozulma çekirdeği veya veri kümesi farklılıklarında performansları düşer. Bu durumlarda sıfır-atış model rekabetçi veya üstün kalabilmekte ve eğitim gerektirmeyen esnek bir seçenek sunmaktadır, fakat daha yüksek çalışma süresi (runtime) maliyetiyle. Bu tez, her iki paradigmayı da ayrıntılı biçimde incelemekte, özellikle zamansal tutarlılık sorununu ele almak için kullanılan farklı yaklaşımlara odaklanmaktadır—göreve özgü modellerde açıkça eğitim tabanlı mekanizmalar, sıfır-atış difüzyon çerçevesinde ise çıkarım zamanı teknikleriyle zamansal tutarlılık sağlama yöntemleri.

ACKNOWLEDGMENTS

I would like to express my deepest gratitude to my advisor, Prof. Ahmet Murat Tekalp, for his invaluable guidance, patience, and inspiration throughout my Ph.D. journey. His mentorship has shaped both my research and the way I approach scientific thinking.

I sincerely thank my thesis committee members—Prof. Yücel Yemez, Assist. Prof. Zafer Doğan, Prof. Gözde Bozdağı Akar, and Prof. Hasan Fehmi Ateş—for their thoughtful feedback and contributions that improved this work.

I am grateful to all members of the Multimedia, Vision, and Graphics Laboratory (MVGL) and KUIS AI Lab for providing an encouraging and collaborative environment. My heartfelt thanks go to Mustafa Akın Yılmaz and Onur Keleş for their friendship, collaboration, and constant encouragement. Special thanks to Codeway Digital Services for their support and for creating a great space to grow both professionally and personally.

I gratefully acknowledge TÜBİTAK for supporting my Ph.D. studies through the project "Image and Video Processing with Deep Learning."

Above all, I am forever thankful to my family, especially my dear sister Roghayeh, for their endless love, patience, and belief in me. Without them, none of this would have been possible. I also wish to express my deepest appreciation to Abbas Samadi, whose presence, kindness, and support made this journey brighter and more meaningful.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS

Abbrev.	Definition
CFA	Cross-Frame Attention
CG	Conjugate Gradient
CLIP	Contrastive Language-Image Pretraining
CNN	Convolutional Neural Network
DAVIS	Densely Annotated VIdeo Segmentation
DDIM	Denoising Diffusion Implicit Model
DDPM	Denoising Diffusion Probabilistic Model
DISTS	Deep Image Structure and Texture Similarity
EDVR	Video Restoration with Enhanced Deformable Convolutional Networks
ERQA	Edge-based Reference-less Quality Assessment
FID	Fréchet Inception Distance
FVD	Fréchet Video Distance
GAN	Generative Adversarial Network
GT	Ground Truth
HR	High Resolution
KL	Kullback–Leibler divergence
LDM	Latent Diffusion Model
LPIPS	Learned Perceptual Image Patch Similarity
LR	Low Resolution
NIQE	Natural Image Quality Evaluator
OF	Optical Flow

Abbrev.	Definition
OF-MSE	Optical Flow Mean Squared Error
PCD	Pyramid, Cascading and Deformable (alignment)
PS	Perceptual Straightness
PSH	Perceptual Straightening Hypothesis
PSNR	Peak Signal-to-Noise Ratio
PVSR	Perceptual Video Super-Resolution
REDS	REalistic and Diverse Scenes
SDXL	Stable Diffusion XL
SISR	Single-Image Super-Resolution
SR	Super-Resolution
SSIM	Structural Similarity Index
TSA	Temporal Spatial Attention
VAE	Variational Autoencoder
VGG	Visual Geometry Group (network)
V1	Primary Visual Cortex
VR	Video Restoration
VSR	Video Super-Resolution

Chapter 1

INTRODUCTION

1.1 Problem and Scope

Many real-world videos are degraded during different processes, including acquisition, compression, storage, and transmission. Restoring these degraded content is fundamentally an ill-posed inverse problem, meaning that multiple plausible high-quality videos may exist for the same degraded observations. Perceptual video restoration addresses this uncertainty by searching solutions that not only satisfy data fidelity but also look natural to humans and remain temporally coherent across frames.

In this dissertation, two perceptual video restoration paradigms are explored:

1. **Specialized learned models**, trained end-to-end for specific task (video super-resolution) with objectives that explicitly encourage natural texture and motion.
2. **Pre-trained large generative models** adapted a latent diffusion model in a zero-shot, training-free manner at inference time for a broad family of video inverse problems (e.g., SR, deblurring, inpainting, motion-blur removal, and their combinations).

We investigate and compare these two paradigms through the lens of the spatio-temporal perception-distortion trade-off: Although fidelity-related objectives often lead to high PSNR/SSIM and perceptual objectives can produce realistic textures, they can cause unnatural or inconsistent motion. A good video restoration must jointly consider spatial appearance and motion coherence.

1.2 Motivation: From Spatial to Spatio-Temporal Perception

The perception-distortion theory [Blau and Michaeli, 2018] is formulated mainly for single images. Extending it to video leads to some challenges: (i) hallucinated textures must evolve consistently over time; (ii) motion should be predictable and physically plausible; and (iii) trade-off evaluation must consider both spatial realism and temporal coherence. Inspired by the perceptual straightening hypothesis [Hénaff et al., 2019] from neuroscience -which suggests that natural videos become more linearly predictable in the early feature spaces of the human visual system - we use trajectory straightness in a perceptual representation to evaluate motion naturalness and guide the design choices in both paradigms.

1.3 Thesis Contributions

This dissertation includes the following contributions.

A. Task-Specific, Perception-Aware VSR

1. **A spatio-temporal perceptual framework** that complements distortion and perceptual measures with a temporal naturalness measure computed in the perceptual feature space. The proposed metrics explicitly connect the motion aspect to the perception–distortion trade-off in video.

2. **A perceptual VSR architecture with dual discriminators** - a spatial (texture) discriminator for realistic detail and a temporal (motion) discriminator that operates on optical flow features - to encourage both natural textures and natural motion. The design is compatible with different VSR backbones and improves temporal perception.

B. Zero-Shot Diffusion for General Video Inverse Problems

1. **Cross-Frame Attention-based temporal reuse** inside a pre-trained latent diffusion pipeline to propagate information across frames without explicit motion estimation.

2. **Perceptual-straightness regularization** which guides sampling trajectories toward smoother temporal paths in perceptual representation, improving temporal naturalness while preserving spatial generative ability.

3. **Multi-path ensemble sampling** for robust, temporally stable reconstructions: we run multiple stochastic diffusion trajectories (sharing the same measurement constraints) and aggregate them to reduce sampling variance, improve overall quality, and suppress frame-to-frame texture flicker.

C. Comparative Analysis Across Paradigms

- **Unified evaluation** on standard video benchmarks (e.g., DAVIS, REDS4) over multiple degradations, reporting spatial fidelity, perceptual image quality, and temporal metrics along with the proposed spatiotemporal measures.

- **Findings:** task-specific models generally achieve stronger fidelity and even perceptual quality when trained for the target task, whereas zero-shot diffusion offers a flexible, training-free solution for different restoration tasks and can be improved with the proposed cross-frame attention, perceptual-straightness guidance, and ensembling.

1.4 Methodological Overview

Part I (Chapter 3) develops a perception-aware VSR approach. We formalize spatiotemporal perception and distortion measures, then introduce a GAN-based architecture with dual discriminators that improves motion naturalness while maintaining realistic textures and show that the introduced measure supports the new architecture design.

Part II (Chapter 4) builds on VISION-XL [Kwon and Ye, 2024] a state-of-the-art video inverse-problem solver which adapts large, pre-trained text-to-image latent diffusion model for video restoration in a zero-shot setting. We propose (i) cross-frame

attention for feature reuse in zero-shot video restoration, (ii) perceptual straightness-guided sampling, and (iii) multi-path ensemble sampling. Together, these inference-time strategies mitigate the temporal inconsistency that typically occurs when image models are applied frame-by-frame and improve the baseline performance.

1.5 Experimental Summary

We evaluate both (i) the task-specific, perception-aware VSR model of Chapter 3 on SR task and (ii) the zero-shot diffusion pipeline of Chapter 4, across diverse restoration tasks (SR, deblurring, inpainting, motion blur, and hybrids). We report distortion metrics (e.g PSNR/SSIM, pixel MSE), perceptual metrics (e.g LPIPS, DISTS, NIQE), temporal metrics (e.g., flow-based MSE, FVD), and the proposed straightness-based measures.

Key findings.

- **Chapter 3 (task-specific VSR).** The perception-aware architecture improves temporal coherence and perceptual quality compared to strong VSR baselines, with a trade-off in distortion. The temporal discriminator and training losses reduce flicker while preserving sharp, natural textures. The improvement in temporal naturalness is also confirmed by proposed straightness measure and visual results.

- **Chapter 4 (zero-shot VSR).** Cross-frame attention (CFA) adds temporal regularization that slightly improves motion coherence in some cases at the cost of minor spatial quality reduction; for first-frame CFA, the impact is anchor-dependent, and windowed variants showed no consistent gains. Perceptual straightness guidance improves the straightness metrics across all tasks and yields FVD gains in some tasks, with small but visible qualitative improvements. Finally, multi-path ensembling leads to the most consistent temporal gains across all the tasks while also improving the spatial realism and fidelity.

- **Comparison across paradigms** Task-specific supervised models are superior when the test conditions match training: they deliver strong fidelity and stable motion at low latency, but require paired data and per-task retraining, and are fragile under dataset distribution or degradation kernel shift. In contrast, zero-shot diffusion models with large generative priors are training-free and task-agnostic, they show stronger robustness to unseen degradations and datasets, but at higher computational cost.

1.6 Thesis Organization

Chapter 2 reviews inverse problems in image/video, classical and deep learning based approaches to restoration, and evaluation metrics, motivating a spatio-temporal perspective. Chapter 3 presents the perception-aware VSR framework (measures, architecture, and training), with quantitative/qualitative evaluation. Chapter 4 improves the zero-shot diffusion-based video restoration pipeline with cross-frame attention, perceptual straightness guidance, and ensemble sampling, including ablations and analysis. Chapter 5 compares two paradigms and concludes with limitations and directions for integrating training-based and zero-shot generative approaches for temporally coherent video restoration.

Chapter 2

BACKGROUND AND LITERATURE REVIEW

2.1 Problem Formulation and Theoretical Foundations

2.1.1 Image and Video Degradation Models

Image degradation refers to the loss of image quality that can occur at any stage of the digital imaging pipeline, from acquisition to final display, including processing, compression, storage, transmission, and reproduction. Deterioration may result from various factors such as noise, compression artifacts, data transmission errors, optical aberrations during acquisition, or inaccurate color reproduction

Given a high-resolution (HR) image $\mathbf{x} \in \mathbb{R}^{w \times h \times c}$, the degradation process results a low-resolution (LR) image $\mathbf{y} \in \mathbb{R}^{\bar{w} \times \bar{h} \times c}$ where $\bar{w} < w$ and $\bar{h} < h$. The relation between HR and LR images can be expressed as follows:

$$\mathbf{y} = D(\mathbf{x}; \Theta) \tag{2.1}$$

where D is the degradation mapping function and Θ is degradation parameters. More generally a "classical degradation model" can be defined as:

$$\mathbf{y} = ((\mathbf{x} \otimes \mathbf{k}) \downarrow_s + \mathbf{n})_q^{\text{JPEG}} \tag{2.2}$$

where $\mathbf{k}$ indicates the blur kernel which models the image-specific motion-blur or camera shake blur, $\otimes$ is convolution operator, s is the downsampling factor, $\mathbf{n}$ represents the additive noise, and, q indicates the JPEG quality factor [Liu et al., 2022][Tekalp et al., 2022].

In standard non-blind SR, the degradation is typically modeled as bicubic downsampling, however, blind SR also includes blur and noise. Other restoration tasks can be implied by considering different realization of this general model. For exam-

ple, deblurring omits downsampling but emphasizes k; denoising emphasizes n; and inpainting models spatial masks within D.

Video degradations adds a frame dependency to the model. For example, for video restoration task the degraded sequence is:

$$V^{\mathrm{LR}}(t) = D\big(V^{\mathrm{HR}}(t); \omega(t)\big)\,, \tag{2.3}$$

where the frame index t lets D and ω to change over time and cause temporal inconsistencies. Additional video degradations include motion blur (object/camera motion), compression artifacts (blocking/ringing), and frame-rate fluctuations that affect temporal continuity.

2.1.2 Restoration as an Ill-Posed Inverse Problem

Image and video restoration including super-resolution (SR), deblurring, denoising, and inpainting, aims to recover a high-quality signal x from a degraded measurement y by trying to invert the forward degradation model $y = D(x;\omega) + n$. In the Hadamard sense, such inverse problems are ill-posed [Tihonov and Arsenin, 1977] meaning that they may lack a solution, violate uniqueness, or be unstable.

The most notable issue in restoration tasks is the violation of uniqueness because the degradation loses information: high frequencies in SR, changes of spectra in blur, missing pixels in inpainting, or corrupted details because of heavy noise/compression. As a result, many distinct solutions for x may map into the identical y.

Therefore, restoration methods use prior knowledge with regularization or Bayesian modeling to deal with ill-posedness. A standard formulation is

$$\hat{x} \;=\; \arg\min_{x} \underbrace{\|y - \mathcal{M}(x)\|^2}_{\text{data fidelity}} + \lambda \underbrace{R(x)}_{\text{prior/regularizer}}\,, \tag{2.3}$$

where $\mathcal{M}$ shows the task-specific degradation model (e.g., blur+downsample for SR, convolution for deblurring, masking for inpainting), and R is the regularizer that encodes prior assumptions about the solution such as smoothness/energy control

(Tikhonov) [Tihonov and Arsenin, 1977], edge preservation, or learned image/video statistics.

2.2 Traditional Restoration Methods

2.2.1 Frequency Domain Approaches for Image Restoration

Frequency-domain approaches support a wide range of restoration methods including super-resolution (SR), deblurring, and denoising. These approaches model the degradations as linear, shift-invariant operators plus noise in the Fourier domain. The leading multi-frame SR work of Tsai and Huang [Tsai and Huang, 1984] utilized the shift property of the Fourier transform to relate multiple aliased low-resolution (LR) observations to a high-resolution (HR) signal via linear equations. Kim et al. [Kim et al., 1990] extended this approach to deal with blur and noise and used the frequency domain formulation for estimations under more realistic degradations.

The same frequency domain principles are used in other restoration tasks: spectral deconvolution for deblurring, frequency-selective attenuation for denoising. However, these approaches usually rely on limiting assumptions such as global translational motion, spatially invariant blur/noise, and accurate degradation models. These assumptions limit their application to real-world degradations such as complex motion inconsistencies, occlusions, spatially varying blur, and compression artifacts. Subsequently, in spite of the strong theoretical foundations and simplicity of frequency-domain methods, their application in general restoration tasks is not sufficient on its own.

2.2.2 Spatial Domain Approaches for Image Restoration

Spatial domain single-image restoration approaches including super-resolution (SR), deblurring, denoising, and inpainting try to recover missing details from a degraded observation by using image priors and explicit modeling of the forward process.

Initial single-image SR methods are interpolation techniques such as bicubic, Lanczos, and spline-based approaches [Keys, 1981]. Although these methods are

computationally efficient,they usually cause artifacts like blurring, jagged edges, and ringing effects. More advanced spatial domain methods leverage image priors to guide the reconstruction process, thereby better preserving structural details.

Edge-directed interpolation [Li and Orchard, 2001] uses upsampling for local edge orientations and improves the sharpness, yet struggles with complex textures. In another work, [Dai et al., 2007] enforces smoothness along edges while preserving discontinuities across them. Gradient-profile priors [Sun et al., 2008] model the shape statistics of natural image gradients to enhance edges during reconstruction. These edge-aware and gradient-based priors are used along with appropriate fidelity term in broader restoration tasks and help stabilize inversion and preserve perceptually important structures.

2.2.3 Regularization with Image Priors

The ill-posed nature of restoration problem necessitates regularization that uses prior knowledge about likely high quality images. The general form of regularized reconstruction can be expressed by Equation 2.3.

Usual priors used for traditional restoration include:

1. **Tikhonov Regularization** [Tihonov and Arsenin, 1977]: modeled the smoothness of natural images using quadratic penalties on the derivatives of restored images:

$$R(\mathbf{x}) \;=\; \|\mathbf{\Gamma x}\|_2^2, \tag{2.4}$$

   where $\mathbf{\Gamma}$ is typically a high-pass filter (e.g., Laplacian). This is an efficient prior usage but can over-smooth the edges/textures.

2. **Total Variation (TV)** [Rudin et al., 1992]: preserves sharp edges by penalizing gradient magnitude,

$$R(\mathbf{x}) \;=\; \|\nabla \mathbf{x}\|_1, \tag{2.5}$$

   performs effective edge-preserving restoration by suppressing small fluctuations while preserving edges,

These objectives are typically solved with different optimization techniques, including gradient descent, conjugate gradients, or iterative reweighted least squares; choosing regularization parameter λ is very important, too small amplifies noise, too large result in over-smoothing of fine details.

2.2.4 Example-Based Methods for Image Restoration

A main shift in *restoration* came with example-based and patch-based methods that exploit redundancy in natural images—either internally (self-similarity) or via external datasets. In SR, Freeman et al. [Freeman et al., 2002] proposed storing corresponding LR–HR patch pairs and enforcing global consistency with a Markov random field, while Chang et al. [Chang et al., 2004] suggests to reconstruct HR patches as linear combinations of similar training exemplars.

Example-based approaches can generate sharp textures and hallucinate details, but performance can be degraded with new patterns, heavy blur, or strong compression artifacts that distort patch similarity. Therefore, they are usually used along with priors or learned components to improve generalization for different restoration tasks.

2.2.5 Traditional Video Restoration

Video restoration (VR) techniques expand image restoration methods by leveraging temporal correlations between consecutive frames. Video restoration methods obtain higher quality reconstructions than single-frame techniques by using the natural motion and redundancy of the video sequences.

Early VR methods used multi-frame image restoration techniques by sliding temporal windows [Zhao and Sawhney, 2002]. They consider neighboring frames as additional observations that need to be accurately registered. For Video Super Resolution (VSR), a typical linear observation model is:

$$\mathbf{y}_t \;=\; \mathbf{D}\,\mathbf{H}\,\mathbf{W}_{t\to r}\,\mathbf{x}_r \;+\; \mathbf{v}_t, \qquad t \in \{r-n,\ldots,r+n\}, \tag{2.6}$$

where $\mathbf{D}$ is downsampling operator, $\mathbf{H}$ is blurring operator, $\mathbf{W}_{t\to r}$ aligns frame t to

the reference r and $2n+1$ is the temporal window size. In other restoration tasks, the same structure is used with the task-specific forward operator. For example, for deblurring $\mathbf{D} = \mathbf{I}$, for denoising $\mathbf{H} = \mathbf{I}$ or for inpainting, it is an additional spatial mask), but accurate warping is used over all classical video restoration tasks.

Farsiu et al. [Farsiu et al., 2004] suggest a multi-frame framework with explicit motion estimation and uses robust norms to reduce outliers due to incorrect registration. This method performs particularly well for sequences with small and well-estimated motions. Similar technique is also effective for tasks like video deblurring and inpainting when alignment is not perfect or there is occlusions.

The main challenge in video restoration is to preserve temporal consistency. Processing frames independently can cause flicker. To solve this problem, early solutions employed temporal regularization [Takeda et al., 2009] to penalize deviations between consecutive reconstructed frames:

$$R_{\text{temp}}(\mathbf{x}_r) \;=\; \lambda_{\text{temp}} \sum_{t \in \{r-1, r+1\}} \|\mathbf{x}_r - \mathbf{W}_{t \to r}\mathbf{x}_t\|_1, \tag{2.7}$$

where λ_{temp} controls the strength of temporal consistency. This idea generalizes naturally to other video restoration tasks, like deblurring and inpainting to reduce flicker and enforce coherence.

Another group of works considers video as a 3D volume and applies spatio-temporal priors directly [Protter et al., 2008]. 3D regularization can preserve temporal coherence, but may over-smooth dynamic content and fine textures. Finally, recursive methods [Vandewalle et al., 2007] use information propagation from previously restored frames, although they risk error accumulation over time, they can improve efficiency. This is a trade-off that is widely used in video restoration tasks.

2.3 Deep Learning based Image Restoration Methods

Traditional restoration methods considered simplified models that could not capture real-world imaging complexities. They need to do careful parameter tuning and can be computationally demanding, which makes general-purpose solutions difficult. Nevertheless, these traditional approaches created valuable mathematical

frameworks and concepts that lead the modern techniques. By arrival of deep learning, the field of image and video restoration has been revolutionized and improved. This section explores the evolution of deep learning-based approaches for single image restoration, investigating their development from convolutional neural networks (CNN) to recent advances in transformers and generative models.

2.3.1 Convolutional Neural Networks

Convolutional neural networks (CNNs) fueled modern restoration by learning data-driven priors in end-to-end manner. The first successful application of deep learning to Single Image Super resolution (SISR) is done by **SRCNN** [Dong et al., 2014], which used a simple three-layer CNN and showed that it could bit the traditional methods. However, SRCNN performs on bicubic-interpolated LR images, making it computationally inefficient. Later works like **FSRCNN** [Dong et al., 2016] did feature extraction on the LR images and used the upsampling operation only at the end of the network. **ESPCN** [Shi et al., 2016] proposed a sub-pixel convolution layer that learned upsampling filters. This significantly reduced computational complexity and improved reconstruction quality. **VDSR** [Kim et al., 2016] showed that increasing the number of layers of CNN could considerably improve the performance. It used a very deep architecture (20 layers) along with global residual learning to facilitate the training.

A significant progress occurred with the introduction of residual learning. **EDSR** [Lim et al., 2017] adapted the residual blocks for the first time, specifically for SR tasks, and created a foundation for later architectures. **SRDenseNet** [Tong et al., 2017] and Residual Dense Network (**RDN**) [Zhang et al., 2018b] provided another breakthrough by using dense blocks with residual connections to support feature reuse and to enable information flow from hierarchical features. These structures are modified and applied in non-SR restoration where rich hierarchical features are equally important.

In summary, although many CNNs were introduced in SR domain, their architectural principles are extended to restoration tasks by modifying the degradation-

aware inputs, data fidelity objectives, and loss functions based on each restoration task.

2.3.2 Transformer-based Approaches

Following the success of transformers in natural language processing (NLP) and computer vision (CV), the attention mechanism has been used for restoration tasks. **IPT** [Chen et al., 2021] was one of the initial works that showed the effectiveness of a transformer-based approach for image processing tasks, including SR. It pre-trained the transformer-based model on large-scale datasets and then fine-tuned it for specific tasks. Swin Transformer is used by **SwinIR** [Liang et al., 2021] for image restoration task. It combines Swin Transformer with a deep feature extraction module and a reconstruction module.

Architectures like **ESRT** [Lu et al., 2021] has been proposed to address the computational demands of transformer-based models and making them more efficient and practical for real-world applications.

2.3.3 Generative Model-based Approaches

A fundamental change occurred in restoration techniques when researchers recognize that pixel accuracy optimization (e.g., pixel space Mean Square Error) does not necessarily reflect perceptual quality. SRGAN [Ledig et al., 2017] used generative adversarial networks (GANs) for SR for the first time. Instead of minimizing pixel-wise error, it used perceptual loss functions which focused on creating photorealistic textures. ESRGAN [Wang et al., 2018] is an improved version of SRGAN which replaced standard residual blocks with Residual-in-Residual Dense Blocks and also adopted a relativistic GAN loss. This strategy improved texture reconstruction and visual quality. The same concepts are used in other restoration tasks such as deblurring, denoising, and inpainting by changing the conditioning and the task-specific fidelity term.

Normalizing-flow methods model the likelihood $p(\mathbf{x} \mid \mathbf{y})$ and enable diverse sampling. SRFlow [Lugmayr et al., 2020] leveraged normalizing flow for SR and made

it possible to generate multiple HR outputs for a single LR input.

Denoising diffusion probabilistic (DDPM) models are state-of-the-art generative approaches for most of the restoration tasks. SR3 [Saharia et al., 2022] adapts a image-to-image diffusion model and does super-resolution through a stochastic iterative denoising process. SRDiff [Li et al., 2022] combined diffusion models with SR-specific conditioning, and achieves high perceptual quality and reasonable reconstruction accuracy.

In spite of GANs, diffusion models have more stable training and usually generate higher quality results, specifically for higher scaling factors or heavier degradations. Diffusion frameworks can be generalized to other restoration tasks by training with task-aware degradations and conditioning.

2.4 Deep Learning based Video Restoration Methods

Unlike single image restoration models, video restoration methods need to deal with motion, occlusions, and long-range dependencies. Different approaches address these challenges with different strategies, such as alignment, temporal propagation, attention, deformable sampling, or generative priors. Despite their differences, they all try to preserve temporally consistency and provide high-quality reconstructions. The following is representative related work for different lines of approaches:

2.4.1 CNN-based Methods

EDVR [Wang et al., 2019] uses pyramid, cascading, and deformable (PCD) alignment along with temporal–spatial attention (TSA) and achieves strong reconstruction quality and robustness to large and complex motion for VSR tasks. **BasicVSR** [Chan et al., 2021] Enhances the pipeline with optical-flow-based alignment and leverages bidirectional feature propagation. This reusing of spatio-temporal features improves consistency and efficiency on long sequences . **BasicVSR++** [Chan et al., 2022] uses second-order deformable propagation and architectural refinement to improve alignment and accuracy for heavy degradations.

2.4.2 GAN-based Methods

Adversarial training improves perceptual quality but if applied per-frame can cause flicker. **TecoGAN** [Chu et al., 2018] uses a spatio-temporal discriminator that processes short frame sequences and a ping–pong temporal consistency loss to reduce drift and inconsistencies. This model restores sharper detail and improves temporal coherence compared to frame-wise GANs.

2.4.3 Diffusion-based Methods

Diffusion models use probabilistic modeling and employ explicit temporal controls to add temporal consistencies. Upscale-A-Video [Zhou et al., 2024], a text-guided VSR diffusion model preserves temporal consistency through incorporating temporal layers inside the UNet and VAE-Decoder and global latent propagation mechanism to ensure temporal coherence. **ZVRD** [Cao et al., 2025]uses a pre-trained single image diffusion model for video in a zero-shot setting by mechanisms like short/long-range temporal attention, consistency guidance, and spatio-temporal noise coordination to achieve temporally coherent restoration. DiffIR2VR-Zero [Yeh et al., 2024] performs zero-shot temporally consistent video restoration by using any pre-trained image diffusion model. It combines hierarchical latent warping with hybrid token merging and maintains temporal consistency while preserving restoration quality for different restoration tasks. **VISION-XL** [Kwon and Ye, 2024] a zero-shot video restoration model utilizes latent diffusion model with pseudo-batch consistent sampling and data consistency refinement to improve temporal coherence and fidelity.

2.5 Quality Assessment and Evaluation Metrics

Different combinations of reference-based distortion measures with no-reference perceptual and temporal-consistency metrics are used in the literature to measure both fidelity and perceptual quality of restored images and videos. Below, we go over the most common metrics that we also utilized in this thesis.

2.5.1 Spatial Fidelity Metrics

MSE (Mean Squared Error) is a full-reference distortion metric that measures the mean squared pixel error between a restored image and ground-truth image. It can prefer blurry outputs that minimize pixel error but perceptually do not look pleasing. It is zero only for a perfect reconstruction and larger values show larger errors.

PSNR [Hore and Ziou, 2010] (Peak-Signal-to-Noise Ratio) is a full-reference distortion metric that shows MSE on a logarithmic (decibel) scale to make error magnitudes easier to compare. It measures pixel-wise accuracy, therefore, higher is better, but similar to MSE it favors pixel-level fidelity over perceptual realism. It is very sensitive to pixel shifts.

SSIM (Structural Similarity Index) [Wang et al., 2004] is a full-reference image quality metric which compares luminance, contrast, and structure statistics within small local windows. In practice, the range is $[0, 1]$ and higher is better. It aligns with human judgments better than MSE/PSNR but still can miss very fine textural realism or hallucinated details that humans might find pleasing.

2.5.2 Perceptual Quality Metrics.

LPIPS (Learned Perceptual Image Patch Similarity) [Zhang et al., 2018a] is a full-reference, perceptual image quality metric. It computes distances in deep feature spaces (e.g., VGG/AlexNet) and normalizes features channel-wise and computes a spatial, channel-weighted distance. Typical values are in $[0, 1]$, lower LPIPS means more perceptually similar (0 for identical images). This metric correlates with human perceptual judgments better than pixel-wise metrics.

NIQE (Natural Image Quality Evaluator) [Mittal et al., 2012] is a no-reference, perceptual image quality metric that computes the deviation statistical features of an image from those of high-quality, "pristine" natural images and lower is better.

DISTS (Deep Image Structure and Texture Similarity) [Ding et al., 2021] is a full-reference, perceptual image quality metric that matches human judgment better than pixel-wise metrics by comparing both structure and texture in a deep feature

space. It is usually reported as a distance and lower means more similar.

ERQA (Edge-based Reference-less Quality Assessment) [Kirillova. et al., 2022] is a no-reference perceptual metric to capture edge sharpness and integrity in restored images. It measures how natural and well-formed the edges are and favors sharp, continuous, and correctly localized edges while penalizing over-smoothing, ringing/halos, double edges, and spurious high-frequency noise.

2.5.3 Temporal Metrics.

OF-MSE (Optical-Flow MSE) is a full-reference temporal measure that measures temporal consistency using optical flow between consecutive frames. A common instantiation warps frame $t-1$ to t with estimated flow for both reference and restored videos and computes an MSE between them. Lower error indicates lower discrepancy between motion fields of ground-truth and restored videos.

FVD (Fréchet Video Distance) [Unterthiner et al., 2019] is a distribution-level metric for evaluating the perceptual quality of videos and it is the video analogue of FID metric for images. It measures how close the distribution of restored videos is to that of real/reference videos in a learned spatio-temporal feature space. It embeds each video with a pre-trained action-recognition network such as I3D on Kinetics to get a feature vector and its empirical means and covariances and then computes the Fréchet distance between these two Gaussians. Lower FVD means restored videos look more like real videos in both spatial content and temporal dynamics. This metric is widely used for perceptual evaluation of video generation/restoration.

Chapter 3

SPATIO-TEMPORAL PERCEPTION-DISTORTION TRADE-OFF VIA TASK-SPECIFIC LEARNED VIDEO SUPER-RESOLUTION MODELS

3.1 Introduction

Video restoration (VR) is one of the main problems in computer-vision, with various applications. Video restoration is challenging because, along with gaining spatial perceptual quality, it requires addressing the additional challenges posed by the temporal dimension. It is necessary not only to reconstruct sharp and natural-looking spatial textures, but also to preserve motion naturalness and temporal coherence between frames. This chapter focuses on specialized video restoration models for the super-resolution (VSR) task (we will investigate generalized pre-trained generative models for video restoration in the next chapter).

Early works on learned video super-resolution employed supervised training to minimize pixel-wise ℓ_2/ℓ_1 loss [Kappeler et al., 2016, Caballero et al., 2017] or minimize the mean-squared error (MSE). However, it is well known that models trained to minimize pixel-wise MSE result in blurry, unnatural-looking textures because the minimum-MSE estimate is a probability-weighted average of all feasible solutions. Later, generative perceptual VSR methods that optimize a weighted combination of ℓ_2/ℓ_1 loss, a no-reference adversarial loss, and full-reference perceptually motivated losses have been proposed.

Typical VSR methods, whether based on fidelity only or perceptual criteria, calculate losses per frame and therefore do not take temporal coherence explicitly into account. Since humans can detect unnatural motion and jitter easily, temporal inconsistencies result in a video with low perceptual quality, even if the texture in

each frame looks natural. Several works explicitly model or enforce the temporal consistency of frames in VSR. However, enforcing naturalness of motion remains a non-trivial problem that should be considered with a new well-defined measure of naturalness of motion.

perception-distortion trade-off [Blau and Michaeli, 2018] is a key concept in image restoration which investigates the trade-off between minimizing pixel-level distortion and maximizing perceptual quality. However, most existing studies have considered this trade-off only in the spatial domain, neglecting challenges related to video tasks, including temporal artifacts such as flicker or jitter that can degrade perceived quality.

This chapter elaborates on our work presented in [Rahimi and Tekalp, 2023], where we extend the perception-distortion framework to the video domain. Specifically, we propose a novel spatio-temporal perceptual quality measure inspired by the *perceptual straightening hypothesis* (PSH) and introduce a new perceptual video super-resolution architecture with dual discriminators to enforce both texture and motion naturalness.

This chapter is organized as follows. Section 3.2 reviews the theoretical background on perceptual VSR (3.2.1) , perception-distortion trade-off (Section 3.2.2), temporal naturalness in video models (Section 3.2.3), and the perceptual straightening hypothesis, grounding it in neuroscience and computational models (Section 3.2.4). Section 3.3 presents our proposed methods, including mathematical formulations for spatio-temporal measures (Sections 3.3.1, 3.3.2) and a spatio-temporal architecture (Section 3.3.3). Section 3.4 discusses our experimental setup, results, and key findings. We conclude in Section 3.5 with reflections on the implications for perceptual video restoration.

3.2 Background

3.2.1 Perceptual VSR

Perceptual VSR models [Lucas et al., 2019, Pérez-Pellitero et al., 2018, Xie et al., 2018, Chu et al., 2020] aim to achieve a trade-off between distortion and perceptual quality using conditional GANs [Qiao et al., 2019] and perceptual losses. [Lucas et al., 2019] employs VSRResNet as the generator along with a discriminator but does not address temporal consistency either in training or in evaluation. [Pérez-Pellitero et al., 2018] proposed a recurrent architecture and a video discriminator to reinforce temporal consistency. Addressing the SR problem for fluid flow, [Xie et al., 2018] employs a temporal discriminator in addition to a spatial discriminator. Beside a recurrent architecture, [Chu et al., 2020] introduced a spatio-temporal discriminator, called TecoGAN, together with a set of training objectives for realistic and temporally coherent VSR.

Motivated by the perceptual straightening hypothesis (PSH) [Hénaff et al., 2019] for human vision, [Kancharla and Channappayya, 2021c] proposes a quality-aware discriminator model to enforce the straightness of the trajectory of the perceptual representations of predicted video frames for the video frame prediction task. In this paper, we apply PSH to the VSR task; furthermore, we do not impose perceptual straightness as a constraint, but use it as a measure for perceptual evaluation of motion.

VSR models are typically evaluated by averages (over frames) of PSNR and SSIM as distortion measures, and of LPIPS and NIQE as perceptual quality measures [Tu et al., 2020]. However, these measures are designed for single images; hence, they do not directly evaluate motion artifacts or temporal coherence of frames. As measures of temporal coherence, [Chu et al., 2020] introduces tOF, which is the pixel-wise difference of estimated flow vectors, and tLP, which is the difference between LPIPS of successive frames of predicted and pristine videos. Alternatively, MOVIE [Seshadrinathan and Bovik, 2009] integrates both spatial and temporal aspects of distortion assessment based on a spatio-spectrally localized multiscale

framework. STEM [Kancharla and Channappayya, 2021a] combines the NIQE metric with a blind temporal algorithm that is based on the perceptual straightening hypothesis [Hénaff et al., 2019]. Nonetheless, there is no study on the effectiveness and role of these measures in evaluating the spatio-temporal perception-distortion trade-off in VSR.

3.2.2 Perception-Distortion Trade-Off

The perception-distortion trade-off, introduced by Blau and Michaeli [Blau and Michaeli, 2018], shows that achieving both low distortion and high perceptual quality is not possible simultaneously. Distortion refers to a full-reference measure quantifying the difference between the ground truth and the reconstructed signal, typically using metrics such as PSNR or SSIM. Perceptual quality, on the other hand, refers to how natural or realistic the output appears to a human observer, often assessed using no-reference metrics like NIQE or full-reference perceptual measures like LPIPS.

While the theory is well established for single-image tasks, its extension to video is non-trivial. The addition of the temporal dimension introduces new forms of perceptual artifacts, particularly those related to motion inconsistency and temporal jitter, which are not captured by frame-based metrics. Addressing this gap requires both new evaluation measures and tailored architectures.

3.2.3 Temporal Consistency and Motion Naturalness

Temporal consistency is the smooth, coherent evolution of content across frames; motion naturalness reflects whether that evolution looks physically and perceptually plausible (predictable) to human observers. Methods that ignore temporal cues often yield perceptually unrealistic motion (e.g., flicker, jitter, shimmer), even when individual frames are sharp and perceptually pleasing.

There is a key distinction between *temporal fidelity to reference motion* and *temporal appearance naturalness*. Flow-based measures such as optical-flow MSE compare motion fields of the restored and reference videos and assess motion fidelity at a coarse structural level. A perceptual video generation/restoration method can

hallucinate fine textures that fluctuate across frames and produce unnatural, unstable appearance—yet achieve a low flow error because the coarse motion matches the reference. In other words, full reference metrics like optical flow MSE does not capture whether hallucinated textures remain temporally consistent, textures might "boil" or fluctuate unnaturally while maintaining correct motion vectors.

Consequently, temporal evaluation should pair motion-fidelity metrics with motion naturalness measures.

3.2.4 Perceptual Straightening in Human Visual Processing

Perceptual straightening is a hypothesis in neuroscience that studied by Hénaff et al. [Hénaff et al., 2019]. They showed that natural video sequences which follow curved trajectories in pixel space become straighter when processed by the hierarchical stages of the human visual system. This transformation makes natural visual sequences more linearly predictable, which is beneficial for anticipating future visual inputs.

To extract the perceptual representation for a video sequence, we use the two-stage nonlinear model suggested by [Hénaff et al., 2019] that simulates the early processing of human visual system illustrated in Figure 3.1.

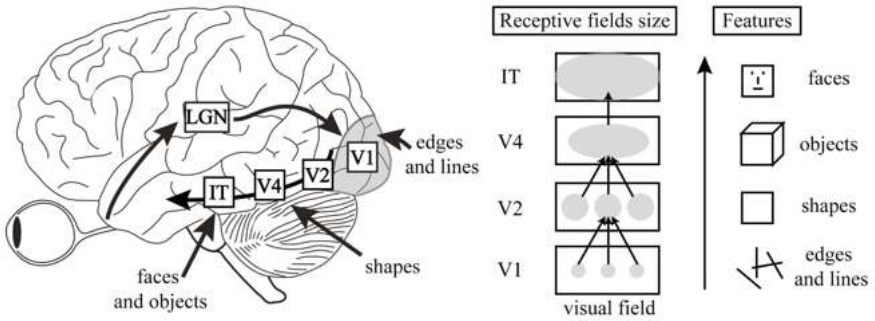

Figure 3.1: Early processing of human visual system [Herzog and Clarke, 2014]

stage1: RetinaND The first stage of the perceptual encoding pipeline is the *RetinaND* module. *RetinaND*is designed to mimic the early visual transformations done in the retina and lateral geniculate nucleus (LGN) of the human's visual system.

This module performs a biologically inspired preprocessing step that transforms raw video frame intensities into representations that are more aligned with human visual systems perception.

The RetinaND module implemented in the official code-base [Hénaff et al., 2019] consists of three convolutional layers, each corresponding to a different operation:

1. **Center-surround filtering**: enhances local contrast and edges via a Difference-of-Gaussians kernel, implemented as a linear convolution:

$$y = k_{\text{lin}} * x \tag{3.1}$$

where x is the input frame, k_{lin} is the center-surround filter, and $*$ shows 2D convolution.

2. **Luminance gain control**: normalizes the filtered response using a smoothed estimate of local brightness:

$$l = k_{\text{lum}} * x, \qquad y \leftarrow \frac{y}{1 + \alpha \cdot l} \tag{3.2}$$

where k_{lum} is the luminance smoothing kernel, and α is a fixed scalar gain parameter.

3. **Contrast gain control**: normalizes the signal based on a local measure of contrast energy:

$$c = \sqrt{k_{\text{con}} * y^2 + \varepsilon}, \qquad y \leftarrow \frac{y}{1 + \beta \cdot c} \tag{3.3}$$

where k_{con} is a contrast kernel, β is another gain parameter, and ε is a small constant for numerical stability.

4. **Nonlinear activation**, applies Soft-plus nonlinearity to enhance numerical robustness and biological plausibility:

$$y \leftarrow \log\left(1 + e^{y}\right) \tag{3.4}$$

All filters and normalization constants are designed analytically using known biological response functions. In spite of its simplicity, RetinaND plays a crucial role in converting the video input into a form that is more aligned with human vision. It normalizes the lighting conditions, emphasizes edges and motion transitions, and subsequently provides a robust representation for subsequent spatiotemporal modeling.

The input and output of RetinaND module for a sample image is presented in Figure 3.2.

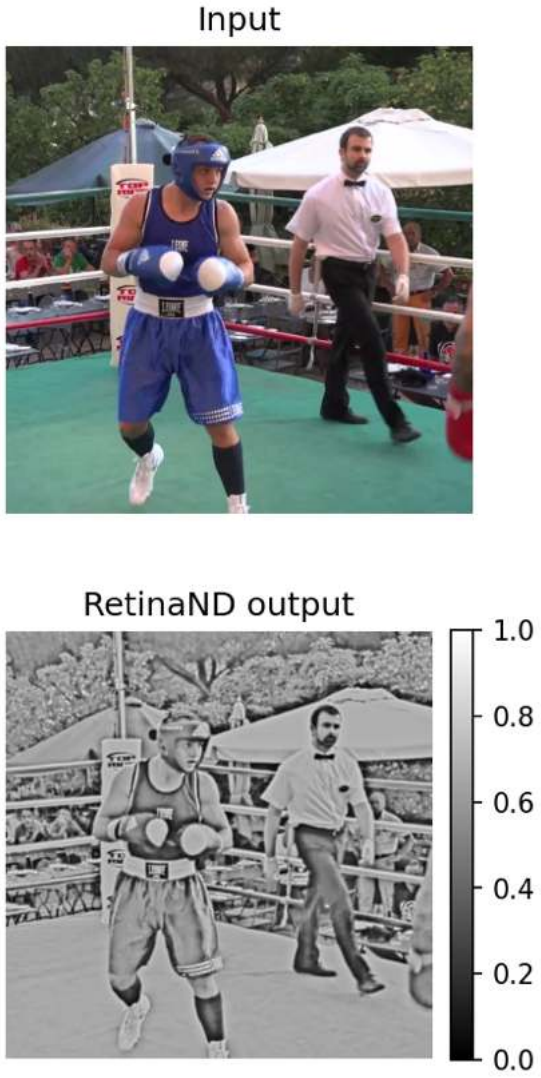

Figure 3.2: Output of RetinaND module

V1 Feature Extraction The second stage of the perceptual processing model mimics the primary visual cortex (V1) which its responses known to be tuned to specific spatial orientations and scales. This stage uses a *Steerable Pyramid* decomposition to extract orientation-, scale-, and frequency-selective responses from the output of the RetinaND module.

The Steerable Pyramid is a linear, multiscale, and orientation-selective transform that uses a set of separable band-pass and low-pass filters in the Fourier domain to extract multi-scale, orientation-selective features through the following steps:

1. The input y is transformed to the frequency domain:

$$\hat{y}^{(0)} = \mathcal{F}\{y\} \tag{3.5}$$

2. At each scale $s = 1, \ldots, S$, band-pass filters $B_k^{(s)}$ extract orientation-selective responses for $k = 1, \ldots, K$ directions, while a low-pass filter $L^{(s)}$ generates the coarser scale:

$$\hat{z}_k^{(s)} = B_k^{(s)} \cdot \hat{y}^{(s-1)}, \qquad \hat{y}^{(s)} = L^{(s)} \cdot \hat{y}^{(s-1)} \tag{3.6}$$

3. Inverse Fourier transform reconstructs the spatial responses:

$$z_k^{(s)} = \mathcal{F}^{-1}\{\hat{z}_k^{(s)}\} \tag{3.7}$$

4. After pyramid decomposition, both the high-pass (finest scale) and low-pass (coarsest scale) bands are removed. The remaining complex-valued orientation responses are converted to magnitude energy maps:

$$e_k^{(s)} = |z_k^{(s)}|^2 = \Re(z_k^{(s)})^2 + \Im(z_k^{(s)})^2 \tag{3.8}$$

5. At last step each energy map $e_k^{(s)}$ is flattened and concatenated into a single feature vector:

$$\mathbf{v} = \texttt{concat}\left(\texttt{vec}(e_k^{(s)}) \;\forall\; s, k\right) \tag{3.9}$$

This final perceptual representation encodes multi-scale, orientation-specific contrast energy, closely reflecting the known response properties of V1 simple cells which is used in the trajectory straightness analysis.

The two-stage perceptual encoder structure is illustrated in Figure 3.3.

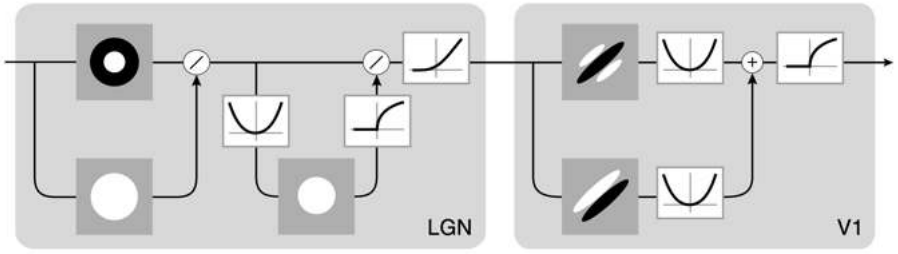

Figure 3.3: Two-stage cascade model describing computations found in the retina, lateral geniculate nucleus (LGN) and V1 [Hénaff et al., 2019]

Application to Video Quality Assessment In the context of video restoration including VSR, the perceptual straightening hypothesis provides a principled way to evaluate temporal naturalness beyond conventional metrics. While metrics such as PSNR or LPIPS focus on spatial fidelity or perceptual similarity between individual frames, they do not account for the coherence of motion across frames.

If we transform a super-resolved video to a trajectory in the aforementioned perceptual representation and then measure how straight (i.e., linearly predictable) that trajectory is, it enables the definition of a *temporal perceptual quality* metric that can better reflect human sensitivity to motion naturalness.

In our proposed work, we integrate perceptual straightness as part of the overall spatio-temporal perceptual evaluation framework, helping to balance the trade-off between spatial realism and temporal coherence in learned video super-resolution models.

3.3 Proposed Methods

We propose a novel framework for addressing the spatio-temporal perception-distortion trade-off in video super-resolution. Our approach combines new evaluation metrics that account for both spatial and temporal perceptual quality with a GAN-based VSR architecture designed to enforce naturalness of both texture and motion.

3.3.1 Spatio-Temporal Perceptual Measure

The spatio-temporal perceptual quality can be evaluated as a combination of spatial and temporal naturalness measures.

Spatial Naturalness: For still frames, it is typical to use LPIPS [Zhang et al., 2018a] or NIQE [Mittal et al., 2012] measures, which estimate the deviation of image statistics from that of natural images. For video, spatial perceptual quality $PQ_{Spatial}$ can be evaluated by averaging LPIPS or NIQE over all frames. For example,

$$PQ_{Spatial} = \frac{1}{N}\sum_{n=1}^{N} LPIPS_{Alex}(X_n, \hat{X}_n) \tag{3.10}$$

Temporal Naturalness: To calculate a measure for temporal naturalness, inspired by the perceptual straightening hypothesis [Hénaff et al., 2019], let's consider an N-frame video $\{X^n\}_{n=1,...,N}$. The perceptual representations $\{P^n\}_{n=1,...,N}$ are high-dimensional vectors in the perceptual space. The curvature $Cur(n)$ at node (frame) n is defined as the angle between successive displacement vectors, $V^n = P^n - P^{n-1}$, $n = 2, 3, ..., N$, which is computed by the dot product of vectors V^n and V^{n+1}

$$Cur(n) = arccos\left(\frac{V^n \cdot V^{n+1}}{\parallel V^n \parallel \parallel V^{n+1} \parallel}\right) \tag{3.11}$$

The straightness of the trajectory at frame n in the perceptual space is given by $ST(n) = \pi - Cur(n)$.

The average straightness of a sequence in the perceptual domain ST^P is defined as a measure of temporal naturalness in degrees.

$$PQ_{Temporal} = \frac{1}{N}\sum_{n=2}^{N}(\frac{ST^P(n)}{\pi} \times 180) \tag{3.12}$$

Spatio-temporal Naturalness: A natural sequence will have higher $PQ_{Temporal}$ and lower $PQ_{Spatial}$. Consequently, a spatio-temporal perceptual quality measure can be defined as

$$P_{ST} = \frac{PQ_{Spatial}}{PQ_{Temporal}} \tag{3.13}$$

where lower scores indicate better quality. While other combinations could be used to fuse the two metrics, we adopt the ratio because it jointly penalizes low spatial perceptual quality and low temporal consistency, and yields lower scores only when both conditions are satisfied.

3.3.2 Spatio-Temporal Distortion Measure

The classical FR measure of fidelity is pixel-wise MSE, which is applied to video by averaging over all N-frames, as

$$MSE_{Pix} = \frac{1}{N}\sum_{n=1}^{N} \|X_n - \hat{X}_n\|_2^2 \tag{3.14}$$

where X_n denotes frame n. To take temporal distortions into account explicitly, we also define optical flow MSE by

$$MSE_{OF} = \frac{1}{N}\sum_{n=2}^{N} \|OF(X_n, X_{n-1}) - OF(\hat{X}_n, \hat{X}_{n-1})\|_2^2 \tag{3.15}$$

where $OF(X_n, X_{n-1})$ is the flow between frames X_n and X_{n-1}. This is similar to t_{OF} measure defined in terms of $l1$ distance [Chu et al., 2020]. We define a spatio-temporal distortion measure D_{ST} for video as a weighted sum of MSE_{Pix} and MSE_{OF}

$$D_{ST} = MSE_{Pix} + \alpha MSE_{OF} \tag{3.16}$$

The parameter α is chosen empirically, considering that the effect of optical flow distortion should neither be neglected nor dominate the overall distortion measure. We set $\alpha = 1000$.

3.3.3 A New Perceptual VSR Architecture For Natural Texture and Motion

We propose a new GAN-based perceptual VSR (PVSR) architecture motivated by the spatial and temporal naturalness measures discussed earlier. The proposed PVSR architecture, depicted in Figure 3.4, employs a generator model, a flow estimation model, and two discriminators. Generator models that minimize a distortion-only loss yield superior fidelity measures, such as PSNR, at the expense of lower perceptual scores. Inclusion of an adversarial texture loss encourages perceptually more realistic textures; however, this causes motion artifacts due to inconsistent hallucinations in successive frames. We show that an additional adversarial motion loss encourages perceptually more natural videos.

PVSR Generator Model The PVSR model can use any generator network G. In this paper, we experimented two VSR models: EDVR [Wang et al., 2019], which processes each picture independent of the previous outputs, hence, is more prone to motion artifacts, and BasicVSR++ [Chan et al., 2022], which is a recurrent model, hence, inherently generates temporally more coherent frames. The resulting two PVSR models are called PEDVR and PBasicVSR++, respectively. Table 3.2 shows that we can get significant improvements in perceptual scores even with PBasicVSR++.

PVSR Texture and Motion Discriminators

To achieve both natural texture and motion, the PVSR model employs two discriminators: a spatial (texture) discriminator D_S to distinguish ground-truth (GT) frames from reconstructed ones, and a temporal (motion) discriminator D_T to discriminate between optical flows estimated from the GT frames and those estimated from reconstructed SR frames. The spatial and temporal discriminators have the same architecture, containing five convolutional layers. The first layer has 3×3 kernel and stride 1, while the others have 4×4 kernels and stride 2. The number of filters in each layer is 64, 64, 64, 128, and 256 respectively. The final feature map is input to a fully connected layer to compute the fake/real scores. The input to the flow discriminator is the optical flow between the current and previous frames estimated by the pre-trained PWC-Net [Sun et al., 2018], depicted with a blue box

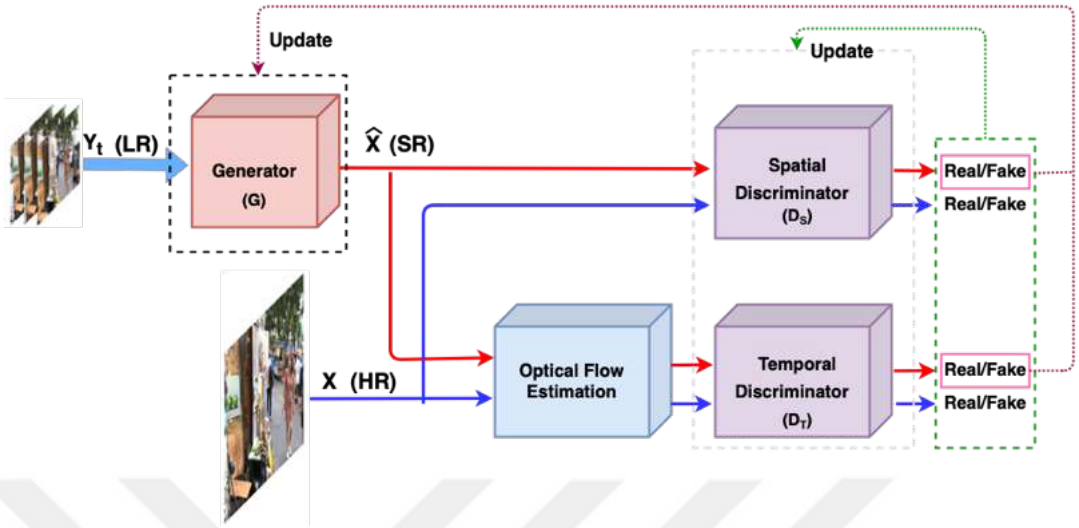

Figure 3.4: The proposed perceptual VSR (PVSR) model architecture. PEDVR and PBasicVSR++ are two instances of PVSR architecture.

in Figure 3.4.

3.3.4 Loss Functions

In order to ensure naturalness of both texture and motion, we propose the generator loss function to include both texture loss $\mathcal{L}_{G,Spatial}$ and motion loss $\mathcal{L}_{G,Temporal}$:

$$\mathcal{L}_G = \mathcal{L}_{G,Spatial} + \mathcal{L}_{G,Temporal} \tag{3.17}$$

Naturalness of Texture: Inspired by SRGAN [Ledig et al., 2017], we use the following texture loss

$$\mathcal{L}_{G,Spatial} = \lambda_1 \mathcal{L}_{Pix} + \lambda_2 \mathcal{L}_{vgg} + \lambda_3 \mathcal{L}_{Pix,adv} \tag{3.18}$$

where $\mathcal{L}_{Pix}$ is the pixel-wise l_2 loss to ensure texture fidelity, $\mathcal{L}_{vgg}$ is l_2 loss of the GT and SR feature maps extracted from a pre-trained VGG-19 network, and the adversarial loss $\mathcal{L}_{Pix,adv}$ aims to maximize the probability that the spatial discriminator will be fooled by the generator given by:

$$\mathcal{L}_{Pix,adv} = -\frac{1}{N}\sum_{n=1}^{N} \log(D_S(\hat{X})) \tag{3.19}$$

where N denotes the number of samples in a mini-batch.

Naturalness of Motion: To ensure that the reconstructed video has coherent motion, the temporal loss is defined by

$$\mathcal{L}_{G,Temporal} = \lambda_4 \mathcal{L}_{Flow} + \lambda_5 \mathcal{L}_{Flow,adv} \tag{3.20}$$

where $\mathcal{L}_{Flow}$ is the l_2 loss between optical flows OF^X and $OF^{\hat{X}}$ estimated from GT frames and SR frames, respectively. $\mathcal{L}_{Flow,adv}$ denotes the adversarial flow loss similar to 3.19, which is calculated based on the output of flow discriminator D_T and encourages the generator to produce sequences with a natural flow. Furthermore, to improve the long-term temporal consistency and avoid temporal accumulation of artifacts, we exploit Ping-Pong loss introduced in [Chu et al., 2020].

3.4 Evaluation

We present the training details, comparison with the state-of-the-art, and an ablation study using different evaluation measures and the perception-distortion trade-off.

3.4.1 Training Details

We trained all models on REalistic and Diverse Scenes (REDS) dataset [Nah et al., 2019], which contains videos with large and complex motions. Similar to other

Table 3.1: Loss terms/weights for each model: Cb, CS, and BCE denote Charbonnier, cosine similarity, and binary cross entropy loss.

	$\mathcal{L}_{Pix}$	$\mathcal{L}_{vgg}$	$\mathcal{L}_{Pix,adv}$	$\mathcal{L}_{Flow}$	$\mathcal{L}_{Flow,adv}$	$\mathcal{L}_{pp}$	$\mathcal{L}_{warp}$	$\mathcal{L}_{offset}$	lr	iter
EDVR [Wang et al., 2019]	Cb / 1	×	×	×	×	×	×	×	4×10^{-4}	600K+600K
PEDVR	l_2 / 1	l_2 / 1	BCE / 0.1	l_2 / 0.2	BCE / 0.1	l_2 / 0.5	×	0.008	5×10^{-5}	50K
Ablation1	l_2 / 1	l_2 / 0.5	BCE / 0.1	×	×	l_2 / 0.5	×	0.008	5×10^{-5}	50K
Ablation2	l_2 / 1	l_2 / 0.5	BCE / 0.1	l_2 / 0.2	×	l_2 / 0.5	×	0.008	5×10^{-5}	50K
BasicVSR++ [Chan et al., 2022]	Cb / 1	×	×	×	×	×	×	×	1×10^{-4}	600K
PBasicVSR++	Cb / 1	l_1 / 1	BCE / 0.05	MSE / 0.1	BCE / 0.05	×	×	×	5×10^{-5}	200K
Ablation1	Cb / 1	l_1 / 1	BCE / 0.05	×	×	×	×	×	5×10^{-5}	200K
Ablation2	Cb / 1	l_1 / 0.5	BCE / 0.05	MSE / 0.1	×	×	×	×	5×10^{-5}	200K
TecoGAN	Cb / 1	CS / 0.2	BCE / 0.01	×	×	Cb / 0.5	Cb / 1	×	5×10^{-5}	500K

Table 3.2: Quantitative comparison of methods on REDS4. The best score is marked blue, while best score within each group is in **bold**.

Method	Varient	PSNR (dB) ↑	SSIM ↑	MSE ↓	NIQE ↓	LPIPS (Alex) ↓	LPIPS (VGG) ↓	OF MSE ↓	Straightness ↑	P-ST↓	D-ST↓
EDVR	EDVR	**30.52**	**0.870**	**65.28**	4.491	0.1982	0.2225	**1.0268**	104.184	0.1090	**75.5500**
	PEDVR (ours)	28.38	0.829	105.08	**2.788**	**0.1064**	**0.1929**	1.1297	**104.454**	**0.0584**	116.3800
	Ablation1 (EDVR)	28.27	0.826	108.97	2.886	0.1075	0.1932	1.1780	104.292	0.0591	120.7600
	Ablation2 (EDVR)	28.29	0.826	107.99	2.876	0.1079	0.1931	1.1270	104.382	0.0592	119.2600
BasicVSR++	BasicVSR++	**32.39**	**0.907**	**43.89**	3.909	0.1302	0.1726	**1.0157**	104.526	0.0714	**54.0470**
	PBasicVSR++ (ours)	31.24	0.897	57.05	**3.193**	**0.0845**	**0.1452**	1.0177	**104.724**	**0.0462**	67.2270
	Ablation1 (BasicVSR++)	31.22	0.886	57.18	3.224	0.0904	0.1497	1.0396	104.544	0.0495	67.5760
	Ablation2 (BasicVSR++)	31.25	0.887	56.98	3.232	0.0861	0.1457	1.0190	104.562	0.0472	67.1700
TecoGAN	TecoGAN	28.13	0.580	111.76	2.959	0.0949	0.1802	1.1350	104.364	0.0521	123.1100

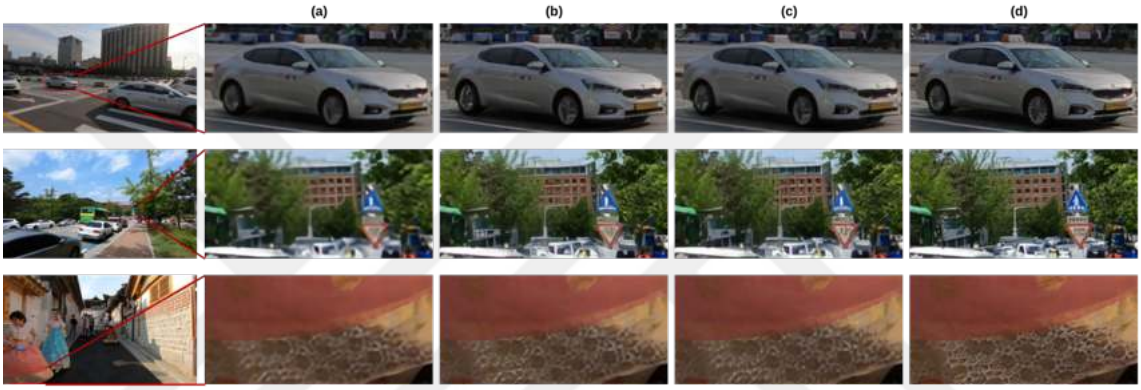

Figure 3.5: Comparison of spatial naturalness using EDVR as generator on REDS4 dataset (a) EDVR, (b) Ablation1, (c) PEDVR , (d) Ground Truth. Please zoom in for better comparison

methods trained on REDS, four clips (000, 011, 015 and 020 - called REDS4) are utilized as the test set, and the remaining 266 clips are used for training. For training, 64×64 patches from the LR frames are randomly cropped and super-resolved with a factor of 4. The Adam optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.99$ is utilized for training. All experiments are performed on a single NVIDIA Tesla V100.

3.4.2 Ablation Studies on PVSR

In order to show the benefit of using the second discriminator and different loss terms on the performance of PEDVR and PBasicVSR++, we conducted two ablations: Ablation 1 model employs only a texture discriminator, trained by the loss function (3.18). Ablation 2 model uses the flow loss in addition to the loss function

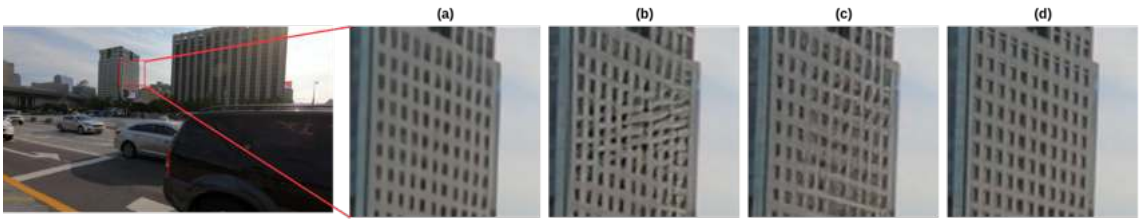

Figure 3.6: motion artifacts, results on REDS4 dataset (a) EDVR, (b) PEDVR, (c) PMEDVR , (d) Ground Truth.

in Ablation 1, and the complete PVSR models employ both discriminators trained by the loss function (3.17).

3.4.3 Comparative Results

We compare the PEDVR model vs. original EDVR and PBasicVSR++ vs. original BasicVSR++. We also compare both PVSR models vs. TecoGAN (as the state-of-the-art temporally coherent perceptual VSR model). For fair comparison, we also trained TecoGAN on the REDS dataset from scratch.

First let's discuss which metrics correlate better with video quality. Clearly, results of BasicVSR++ (evaluated as still frames) look more natural than those of EDVR; yet, EDVR results have better NIQE scores than BasicVSR++. This is because the NR measure NIQE improves as the amount of hallucinated texture increases regardless of fidelity, and the recurrent framework of BasicVSR++ limits unrestricted hallucination in successive frames. Hence, we decided to use LPIPS in P_{ST} rather than NIQE because we believe LPIPS is better correlated with the quality of VSR frames. In terms of naturalness of motion, we observe that the Straightness measure correlates well with visual quality and OF MSE.

Inspection of Table 3.2 shows that both EDVR and BasicVSR++ (optimized solely on pixel-wise Charbonnier loss) achieve better PSNR and OF MSE, while NIQE and LPIPS scores are worse compared to their PVSR counterparts.

In both group of experiments, comparing to the plain models, Ablation 1 models aim to obtain natural texture using adversarial texture and VGG losses that is re-

flected in lower NIQE/LPIPS as well as sharper and more natural texture illustrated in Figure 3.5. However, the flow fidelity is also deteriorated since texture adversarial loss leads to inconsistent textures in successive frames and tangible motion artifacts as depicted in Figure 3.6.

Introducing $\mathcal{L}_{Flow}$ term in the loss function leads to lower OF MSE in Ablation 2 models, while introducing the second (motion) discriminator leads to the best LPIPS and straightness scores achieved by the proposed PVSR models.

We also note that PBasicVSR++ model is clearly superior to TecoGAN (state-of-the-art temporally coherent perceptual VSR model) and PEDVR in all mentioned measures due to two reasons: First, we employ two separate discriminators to allow learning spatial or temporal naturalness distributions separately as opposed to a single spatio-temporal discriminator. Second, BasicVSR++ is a stronger generator model compared to EDVR and the FRVSR network used in TecoGAN.

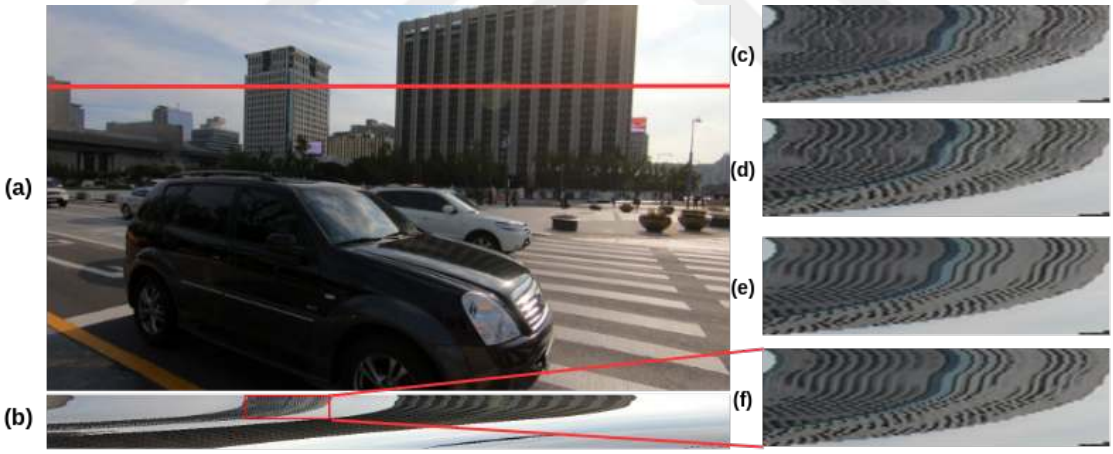

Figure 3.7: (a) Video from REDS4 dataset, (b) Ground-truth temporal profile for the red line; (c)-(f) zoom in to red box for: (c) Ablation1 (PEDVR), (d) PEDVR, (e) PBasicVSR++, (f) Ground-truth.

In order to demonstrate the naturalness of motion, Figure 3.7 depicts temporal profiles extracted from 100 sequential frames. Comparison of zoom-ins in (c)-(f) shows the superiority of PBasicVSR++ model over all other models, and also that PEDVR has less temporal artifacts compared to Ablation 1 model, which does not use temporal losses.

Finally, the spatio-temporal perception-distortion trade-off can be evaluated in terms of the proposed D_{ST} and P_{ST} measures, which capture the spatio-temporal distortion and perceptual quality of a video. Table 3.2 shows that the original EDVR and BasicVSR++ models have the best D_{ST} and worst P_{ST} compared to their PVSR counterparts and their ablations. While, the combination of losses in 3.17 enables proposed PVSR models to gain the best P_{ST} and strike the desired spatio-temporal perception-distortion trade-off compared to the original models as well as their ablation models.

3.5 Summary and Conclusion

It is well-accepted in the community that there is no single measure of image/video quality that correlates well with human preferences. Furthermore, commonly used image perception measures such as LPIPS and NIQE, do not reflect naturalness of motion in videos. To this effect, we propose the perceptual straightness as a measure of motion naturalness and also propose a new PVSR model with two discriminators, where a flow discriminator encourages naturalness of motion. As a result, this paper advances the state-of-the-art in spatio-temporal perception-distortion tradeoff in VSR.

Takeaways i) a strong generator model is the most important factor to obtain the best perceptual results, ii) the perceptual quality/scores for the best model (BasicVSR++) can still be significantly improved by our PVSR architecture, iii) NIQE scores do not correlate well with visual VSR quality, iv) perceptual straightness measure correlates well with motion naturalness, and v) the second (motion) discriminator improves the straightness scores.

Limitations Our approach has the problems of GAN-style training (stability and tuning) and depends on the quality of estimated optical flow : imperfect optical flow can bias the temporal signal, and additional compute is required for the temporal discriminator at training time. Generalization outside the trained degradation

regimes is not guaranteed, especially under compound or unseen artifacts.

Next chapter outlook The next chapter investigates a zero-shot training-free generative direction for general video restoration. We use VISION-XL as the baseline and replace training-time temporal objectives used in this chapter with inference-time mechanisms such as cross-frame attention for information reuse, perceptual straightness-guided sampling as a temporal prior, and multi-path ensembling to suppress stochastic texture jitter

Chapter 4

ENFORCING TEMPORAL CONTINUITY VIA PRE-TRAINED LARGE GENERATIVE MODELS

4.1 Introduction

This chapter explores a training-free, zero-shot video restoration paradigm based on the VISION-XL framework [Kwon and Ye, 2024]: a state-of-the-art latent diffusion pipeline adapted for video inverse problems. This paradigm recovers a clean video from degraded observations under a known (or estimated) forward model by using an image-based pre-trained diffusion model at inference time rather than re-training a video model for each task.

Although VISION-XL implicitly forces temporal consistency through different mechanisms, it is essentially a frame-based framework and lacks explicit modeling of temporal dynamics or motion-aware priors. This limitation motivates us to explore temporally-aware extensions, presented in the next section. Specifically, we augment VISION-XL with three inference-time mechanisms that target temporal stability while preserving spatial realism: (i) cross-frame attention to reuse information across neighboring frames without explicit motion estimation. (ii) perceptual straightness guidance that steers diffusion sampling toward smoother temporal trajectories. (iii) multi-path ensembling that aggregates multiple stochastic diffusion trajectories to suppress frame-to-frame texture jitter. In chapter 3 we enforce the temporal naturalness via training-time losses for a task-specific VSR model, while in this chapter we use inference-time mechanisms in a large-scale pre-trained diffusion prior. The former offers high fidelity and low latency for its target task, while the latter trades computation for flexibility across a wide range of degradations without additional training.

Chapter structure. This chapter is organized as follows. **Section 4.1** presents the introduction. **Section 4.2**, establishes the methodological background (4.2.1). It introduces diffusion probabilistic models, Latent Diffusion Models, narrows to latent diffusion (SDXL), formulates diffusion-based posterior sampling for inverse problems, introduces VISION-XL as the zero-shot baseline, and situates our goals within prior work on temporal consistency **Section 4.3** to **Section 4.5** each develops an inference-time mechanism within VISION-XL pipeline: Section 4.3 motivates and defines cross-frame attention, starting from attention fundamentals and its use inside diffusion U-Nets before detailing our cross-frame variants; Section 4.4 introduces perceptual straightening within VISION-XL and outlines the optimization strategies used to bias trajectories toward smoother temporal evolution; Section 4.5 presents multi-path ensemble sampling, analyzing sources of stochasticity, specifying the ensemble procedure, and providing a brief theoretical rationale. Section 4.6 reports the experimental evidence: it first specifies the evaluation setup (datasets, degradations, and metrics), then records the baseline performance of VISION-XL, and finally isolates the contribution of each mechanism and the ablations.

4.2 Background

4.2.1 Foundations of Diffusion Probabilistic Models

Diffusion Probabilistic Models (DPMs), specifically Denoising Diffusion Probabilistic Models (DDPMs) [Ho et al., 2020], are a group of generative models that have demonstrated remarkable performance in high-fidelity image generation and now form the backbone of several advanced generative methods [Dhariwal and Nichol, 2021]. Diffusion models generally include two processes: the forward process, that gradually adds noise to the data sample based on a Markovian chain, and a reverse process that reconstructs the clean data from the noise.

Forward Process. Let $\mathbf{x}_0 \sim p_{\text{data}}(\mathbf{x})$ be a data sample from distribution of p_{data}. In the forward or diffusion process, a Gaussian noise is gradually added over T

discrete timesteps, forming a noisy data trajectory $\mathbf{x}_0 \to \mathbf{x}_1 \to \cdots \to \mathbf{x}_T$ defined as:

$$q(\mathbf{x}_t|\mathbf{x}_{t-1}) := \mathcal{N}(\mathbf{x}_t; \sqrt{1-\beta_t}\mathbf{x}_{t-1}, \beta_t\mathbf{I}), \tag{4.1}$$

where $\{\beta_t\}_{t=1}^T$ is a fixed variance schedule that controls the noise amount. A good property of the forward process is that we can sample $\mathbf{x}_t$ at an arbitrary timestep t from $\mathbf{x}_0$ directly using the following formula which is the closed form of iterative application of (4.1):

$$q(\mathbf{x}_t|\mathbf{x}_0) = \mathcal{N}(\mathbf{x}_t; \sqrt{\bar{\alpha}_t}\mathbf{x}_0, (1-\bar{\alpha}_t)\mathbf{I}), \tag{4.2}$$

where $\bar{\alpha}_t := \prod_{s=1}^t (1-\beta_s)$.

Reverse Process. If we can reverse the forward process and sample from $q(\mathbf{x}_{t-1}|\mathbf{x}_t)$, we will be able to generate the clean sample from a Gaussian noise input, However, calculating $q(\mathbf{x}_{t-1}|\mathbf{x}_t)$ requires knowing the joint distribution over the entire dataset which is not feasible and therefore we need to learn a model to approximate these conditional probabilities in order to run the reverse diffusion process:

$$p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) := \mathcal{N}(\mathbf{x}_{t-1}; \boldsymbol{\mu}_\theta(\mathbf{x}_t, t), \Sigma_\theta(\mathbf{x}_t, t)). \tag{4.3}$$

Experimentally, [Dhariwal and Nichol, 2021] choose to set $\Sigma_\theta(\mathbf{x}_t, t) = \beta_t\mathbf{I}$ while the mean $\boldsymbol{\mu}_\theta$ is obtained by predicting the noise $\boldsymbol{\epsilon}$ added in the forward process. One common objective is the simple mean-squared error (MSE) loss between the true and predicted noise:

$$\mathcal{L}_{\text{simple}} = \mathbb{E}_{\mathbf{x}_0,\boldsymbol{\epsilon},t}\left[\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)\|^2\right]. \tag{4.4}$$

The forward and backward processes are shown in figure 4.1.

Deterministic Sampling and DDIM Inversion. Denoising Diffusion Implicit Models (DDIM) [Song et al., 2020] suggests a non-Markovian generalization of the DDPM reverse process that enables deterministic sampling while leading to the same marginal distributions $q(x_t|x_0)$. The DDIM reverse process is formulated as:

$$x_{t-1} = \sqrt{\bar{\alpha}_{t-1}} \cdot \text{pred}_{x_0} + \sqrt{1-\bar{\alpha}_{t-1}-\sigma_t^2} \cdot \epsilon_\theta(x_t, t) + \sigma_t\epsilon \tag{4.5}$$

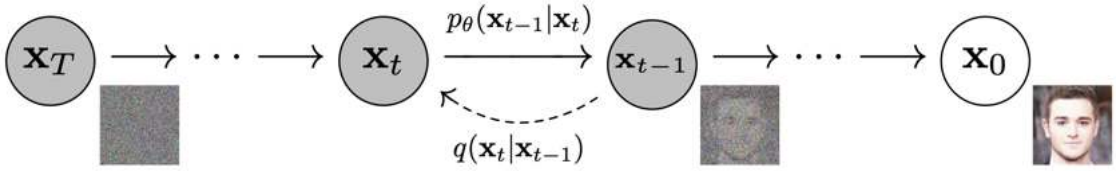

Figure 4.1: The Markov chain of forward diffusion process of generating a sample by step-by-step adding noise and the reverse process of gradually estimating and removing the noise [Dhariwal and Nichol, 2021]

where pred_{x_0} is the predicted clean image, and σ_t is a parameter that controls the stochasticity of the process.

Clean image predicted at step t, is computed using Tweedie's formula [Efron, 2011]. as below:

$$\text{pred}_{x_0} = (x_t - \sqrt{1-\bar{\alpha}_t}\epsilon_\theta(x_t, t))/\sqrt{\bar{\alpha}_t} \tag{4.6}$$

When $\sigma_t = 0$, we have deterministic sampling that enables *DDIM inversion*, in which a clean image x_0 can be mapped to its corresponding noise x_T by reversing the sampling trajectory by iteratively applying:

$$x_t = \sqrt{\bar{\alpha}_t} \cdot \text{pred}_{x_0} + \sqrt{1-\bar{\alpha}_t} \cdot \epsilon_\theta(x_{t-1}, t-1) \tag{4.7}$$

DDIM inversion provides a bijective mapping between the data and noise manifolds and widely used in applications such as image editing, where we can invert a real image to noisier space, and then denoise it using necessary conditions to have an edited version.

4.2.2 Latent Diffusion Models

Although pixel-space diffusion models [Ho et al., 2020, Dhariwal and Nichol, 2021] have achieved great success in generative applications , their use in high-resolution image synthesis and related tasks is limited by computational complexity and memory requirements. These limitations are due to the need to iterative sampling in

high-dimensional pixel space $\mathbb{R}^{H\times W\times 3}$. To overcome these limitations, *Latent Diffusion Models* (LDMs) [Rombach et al., 2022a] suggests to encode the pixel space into a latent representation and apply diffusion processes in this more compact space.

Theoretical Formulation. Let's consider $\mathbf{x}_0 \in \mathbb{R}^{H\times W\times C}$ is the clean image, and (E_θ, D_θ) denotes a pretrained encoder–decoder pair, then clean image $\mathbf{x}_0$ and ist corresponding latent representation $\mathbf{z}_0$ are connected by:

$$\mathbf{z}_0 = E_\theta(\mathbf{x}_0), \quad \hat{\mathbf{x}}_0 = D_\theta(\mathbf{z}_0), \quad \text{with } \mathbf{z}_0 \in \mathbb{R}^{h\times w\times c}, \quad h \ll H,\ w \ll W. \tag{4.8}$$

In practice, a variational autoencoder (VAE) objective is used to train the encoder-decoder networks, E_θ and D_θ . Usually, this objective minimizes a weighted sum of pixel-space reconstruction error and a KL divergence regularizer [Ho et al., 2020, Dhariwal and Nichol, 2021] . The latent space of a trained VAE becomes a compact, approximately Gaussian-distributed embedding space, that can be used in generative modeling.

A diffusion model is then trained in $\mathcal{Z}$ using the same formalism, substituting $\mathbf{z}_t$ with the image variable $\mathbf{x}_t$:

$$q(\mathbf{z}_t|\mathbf{z}_{t-1}) = \mathcal{N}(\mathbf{z}_t; \sqrt{1-\beta_t}\mathbf{z}_{t-1}, \beta_t\mathbf{I}) \tag{4.9}$$

$$q(\mathbf{z}_t|\mathbf{z}_0) = \mathcal{N}(\mathbf{z}_t; \sqrt{\bar{\alpha}_t}\mathbf{z}_0, (1-\bar{\alpha}_t)\mathbf{I}) \tag{4.10}$$

$$p_\theta(\mathbf{z}_{t-1}|\mathbf{z}_t) = \mathcal{N}(\mathbf{z}_{t-1}; \boldsymbol{\mu}_\theta(\mathbf{z}_t, t), \Sigma_\theta(\mathbf{z}_t, t)). \tag{4.11}$$

Sampling is done iteratively in the latent space $\mathcal{Z}$, and only the final estimate $\mathbf{z}_0$ is decoded to pixel space using D_θ.

LDMs offer significant advantages over pixel-space diffusion models by operating in a lower-dimensional latent space $\mathcal{Z}$. This leads to notable computational and memory efficiency because of the reduced spatial resolution often 1/8 to 1/16 of the original image. The VAE encoder E_θ also provides semantic compression by eliminating pixel space redundancies while preserving essential structural information. Additionally, the pretrained encoder-decoder can be used in different diffusion models in plug-and-play form and allows different extensions such as class- or text-conditional generation without need to change the latent representations.

Stable Diffusion and SDXL. Stable Diffusion [Rombach et al., 2022a] and its successor SDXL [Podell et al., 2023] are two most successful latent diffusion models for text-to-image synthesis. A general LDM structure is shown in figure 4.2. VAE Encoder-Decoder pair (E_θ, D_θ), a denoising network (ϵ_θ), and, a condition encoder (τ_θ) are the main components of this structure. Stable Diffusion uses a UNet-based denoising backbone and a pretrained CLIP-based text encoder to do text-conditioned image generation. SDXL extends this structure by introducing a hierarchical latent structure, improved attention mechanisms, and enhanced VAE architectures. The UNet architecture is enlarged to 2.6B parameters (3× larger than SD 1.5) with deeper transformer blocks and enhanced cross-attention layers that better capture semantic relationships across spatial regions. Also, SDXL uses conditioning refinements including resolution-aware training to handle different multiple aspect ratios.

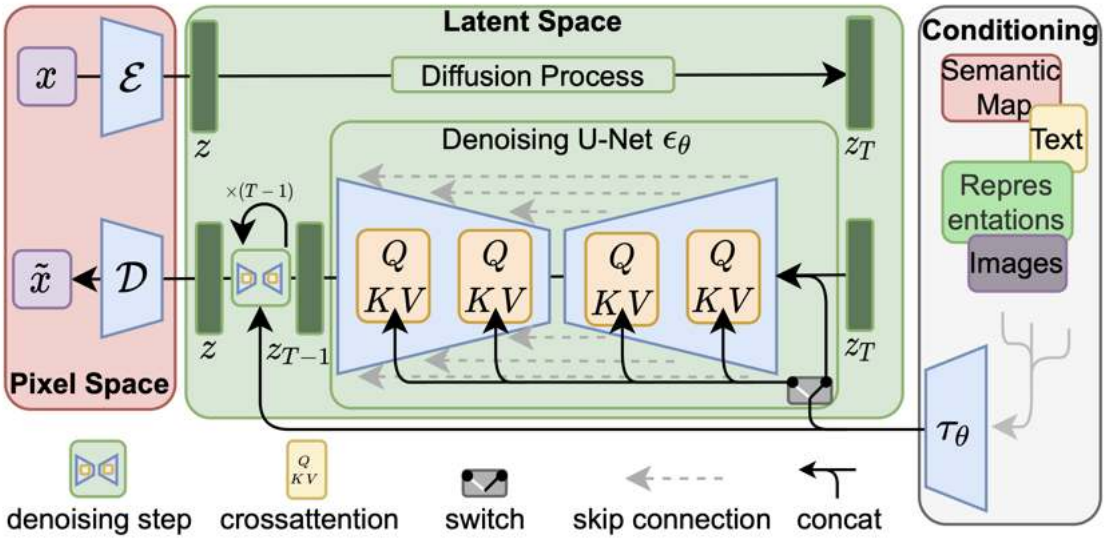

Figure 4.2: Latent Diffusion Model [Rombach et al., 2022a]

The aforementioned foundations are the required basis to adapt the diffusion models in inverse problems like video restoration. The next subsections build upon this theoretical basis to explore specific inverse modeling strategies and the baseline architecture employed in our work.

4.2.3 Diffusion Models for Inverse Problems

Inverse problems include estimating an unknown signal $\mathbf{x}_0$ from degraded or partial observations (measurement) $\mathbf{y} = A(\mathbf{x}_0)$, where $A : \mathcal{X} \to \mathcal{Y}$ is the data corruption process. Classical approaches use hand-crafted priors or learned mappings; however, generative models, especially diffusion-based ones offer a probabilistic framework to solve inverse problems by leveraging learned data priors $p_\theta(\mathbf{x})$ [Chung et al., 2022].

Bayesian Posterior Sampling. Considering the joint distribution of ground-truth and the corresponding degraded signal $p_\theta(\mathbf{x}_0, \mathbf{y})$, the restoration task can be interpreted as sampling from the posterior distribution $p_\theta(\mathbf{x}_0|\mathbf{y})$. Based on Bayes' rule:

$$p_\theta(\mathbf{x}_0|\mathbf{y}) \propto p_\theta(\mathbf{x}_0) \cdot p(\mathbf{y}|\mathbf{x}_0), \tag{4.12}$$

where $p_\theta(\mathbf{x}_0)$ is the prior which models the image structure and can be learned by diffusion model, and $p(\mathbf{y}|\mathbf{x}_0)$ is the likelihood that uses degradation model A to encode the consistency with the measurement. In diffusion-based generative models, $p_\theta(\mathbf{x}_0)$ is implicitly estimated via reverse diffusion process over noisy intermediate variables $\mathbf{x}_t$, and the main problem is how to incorporate the conditioning from $p(\mathbf{y}|\mathbf{x}_0)$ into this generative process.

Conditional Diffusion Framework. In order to sample from the posterior $p_\theta(\mathbf{x}_0|\mathbf{y})$ within the diffusion sampling framework, several strategies have been proposed in the literature. Let $\mathbf{x}_t$ denote the noisy sample at timestep t in the reverse chain.

A practical approach is to guide the diffusion sampling (reverse process) via intermediate denoised samples $\hat{\mathbf{x}}_0^{(t)}$ (pred_{x_0}), estimated using Tweedie's formula:

$$\hat{\mathbf{x}}_0^{(t)} = \frac{1}{\sqrt{\bar{\alpha}_t}} \left(\mathbf{x}_t - \sqrt{1 - \bar{\alpha}_t} \cdot \boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t) \right), \tag{4.13}$$

and then refine $\hat{\mathbf{x}}_0^{(t)}$ to minimize a data-fidelity loss with respect to the observation (measurement):

$$\mathcal{L}_{\text{data}}(\hat{\mathbf{x}}_0^{(t)}) = \|A(\hat{\mathbf{x}}_0^{(t)}) - \mathbf{y}\|^2. \tag{4.14}$$

The refined $\hat{\mathbf{x}}_0^{(t)}$ is then re-noised to continue the sampling.

4.2.4 Baseline: VISION-XL for Zero-Shot Video Inverse Problems

VISION-XL [Kwon and Ye, 2024] is a state-of-the-art video inverse problem solver framework that uses pretrained SDXL [Podell et al., 2023] in a zero-shot manner. It proposes algorithmic strategies to adapt single-frame latent diffusion models to multi-frame video restoration tasks. Figure 4.3 illustrates the framework used in VISION-XL.

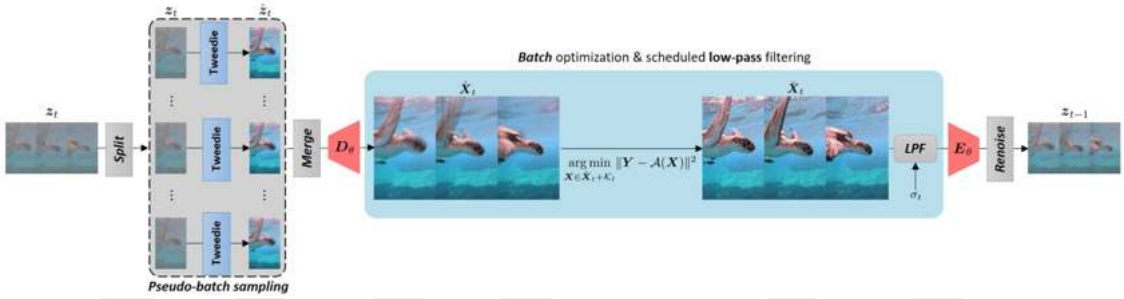

Figure 4.3: Overview of the VISION-XL pipeline [Kwon and Ye, 2024].

Problem Setup. Given a degraded video sequence $\{\mathbf{y}^{[n]}\}_{n=1}^{N}$ with N frames and a known degradation mapping function $A : \mathbb{R}^{H\times W\times 3} \to \mathbb{R}^{H'\times W'\times 3}$ (explained in 2.1.1), the goal is to restore high-quality frames $\{\mathbf{x}_0^{[n]}\}_{n=1}^{N}$ from degraded frames such that:

$$\mathbf{x}_0^{[n]} \sim p_\theta(\mathbf{x} \mid \mathbf{y}^{[n]}) \propto p_\theta(\mathbf{x}) \cdot p(\mathbf{y}^{[n]} \mid A(\mathbf{x})), \tag{4.15}$$

where $p_\theta(\mathbf{x})$ is the image prior modeled via the pretrained latent diffusion model and $p(\mathbf{y}^{[n]} \mid A(\mathbf{x}))$ is the likelihood that should be incorporated in the diffusion sampling process.

Latent Diffusion Initialization via DDIM Inversion. In order to generate consistent initializations across frames, instead of initializing all frames with pure Gaussian noise, Vision-XL uses DDIM inversion (4.2.1). The initial latent sequence

$\mathbf{Z}_\tau$ is constructed by first encoding the first degraded video frame $\mathcal{Y}^{[1]}$ to latent space using the VAE encoder E_θ, then applying DDIM inversion to compute a noisy latent $\mathbf{z}_\tau$, which is repeated across all frames:

$$\mathbf{z}_{\mathcal{Y}}^{[1]} = E_\theta(\mathcal{Y}^{[1]}), \quad \mathbf{z}_\tau = \text{DDIM}^{-1}(\mathbf{z}_{\mathcal{Y}}^{[1]}), \quad \mathbf{Z}_\tau := [\mathbf{z}_\tau, \ldots, \mathbf{z}_\tau] \tag{4.16}$$

where DDIM^{-1} denotes DDIM inversion. This avoids random initialization and at the same time improves data-fidelity and the cross-frame temporal consistency.

Estimating Clean Image. At each timestep $t \in [\tau, \ldots, 0]$, the noisy latent $\mathbf{z}_t$ corresponding to each individual frame, along with sampling timestep t and null prompt embeddings are fed into the pretrained SDXL UNet to predict the noise $\boldsymbol{\epsilon}_\theta(\mathbf{z}_t, t)$. Then, the intermediate clean latent is estimated using Tweedie's formula for each individual frame in parallel:

$$\hat{\mathbf{z}}_0 = \frac{1}{\sqrt{\bar{\alpha}_t}} \left(\mathbf{z}_t - \sqrt{1 - \bar{\alpha}_t} \cdot \boldsymbol{\epsilon}_\theta(\mathbf{z}_t, t) \right). \tag{4.17}$$

Data Consistency Optimization. After obtaining a clean latent estimate $\hat{\mathbf{z}}_0$, VISION-XL performs pixel-space optimization to increase the likelihood $p(\mathbf{y}|\mathbf{x}_0)$ and as a result, to enforce data fidelity. To do this, the estimated latent is decoded to obtain clean frame $\hat{\mathbf{x}}_0 = D_\phi(\hat{\mathbf{z}}_0)$. Then the estimated clean frames are refined by minimizing a data-consistency loss using l-step conjugate gradient (CG) optimization:

$$\bar{\mathbf{x}}_0 := \arg \min_{X \in \hat{X}_0 + \mathcal{K}_l} \|Y - A(X)\|_2^2 \tag{4.18}$$

where $\mathcal{K}_l$ denotes the l-dimensional Krylov subspace associated with the given inverse problem [Chung et al., 2023].

Low-Pass Filtered Encoding. This step applies a scheduled Gaussian filter to the CG-optimized $\bar{x}_0$ before re-encoding to latent space. The filter width is set proportional to the noise scale: $\sigma_t = \lambda\sqrt{1 - \bar{\alpha}_t}$. This low-pass filter mitigates VAE error accumulation during iterative encode-decodes and removes high-frequency artifacts that emerge from the optimization process.

Renoising The refined and filtered frames are then re-encoded to latent space and re-noised into $\mathbf{z}_{t-1}$ to continue the sampling.

$$\begin{aligned} \bar{\mathbf{z}}_0 &= E_\phi\left(\bar{\mathbf{x}}_0\right) \\ \mathbf{z}_{t-1} &= \sqrt{\bar{\alpha}_{t-1}}\bar{\mathbf{z}}_0 + \sqrt{1-\bar{\alpha}_{t-1}} \cdot \boldsymbol{\epsilon}_t, \end{aligned} \tag{4.19}$$

where $\boldsymbol{\epsilon}_t$ is composed of batch-consistent noise and deterministic noise:

$$\begin{aligned} c_1 &= \sqrt{1-\bar{\alpha}_{t-1}} \cdot \eta \\ c_2 &= \sqrt{1-\bar{\alpha}_{t-1}} \cdot \sqrt{1-\eta^2} \\ \mathbf{z}_{t-1} &= \sqrt{\bar{\alpha}_{t-1}}\bar{\mathbf{z}}_0 + c_1 \cdot \boldsymbol{\epsilon}_{\text{fixed}} + c_2 \cdot \boldsymbol{\epsilon}_\theta(\mathbf{z}_t, t) \end{aligned} \tag{4.20}$$

In summary, VISION-XL framework adapts SDXL, a powerful image-based latent diffusion model for zero-shot video restoration using carefully designed inference-time mechanisms. Although it implicitly forces temporal consistency through batch-consistent initialization, or fixed noise, it is essentially a frame-based framework and lacks explicit modeling of temporal dynamics or motion-aware priors. This issue limits its ability to fully capture inter-frame dynamics. These limitations motivate us to explore temporally-aware extensions, presented in the next section.

4.2.5 Temporal Consistency in Video Restoration with Diffusion Models

Temporal coherence across frames plays a great role in perceptual quality of a restored video. Although text-to-image diffusion models have achieved remarkable results in single image generation, their per-frame application to video leads to temporal inconsistencies such as flickering.

Problem Formulation. Let $\{y^{[n]}\}_{n=1}^N$ denote a degraded video sequence and $\{\hat{x}_0^{[n]}\}_{n=1}^N$ be the estimated clean video sequence. The image-based diffusion approaches handles restoration task as an independent optimization problem per frame:

$$\hat{x}_0^{[n]} = \arg\max_x p_\theta(x) \cdot p(y^{[n]} \mid A(x)^{[n]}), \quad \forall n \in \{1, \ldots, N\}, \tag{4.21}$$

This independent processing fails to capture the inter-frame dependencies, making the result temporally incoherent.

Vision-XL Temporal Consistency Mechanisms. Although VISION-XL is based on image-based diffusion model and does not use explicit temporal modeling, it leverages several inference-time strategies to improve temporal coherence across video frames:

- **Consistent DDIM Initialization:** All frames are initialized using a deterministic DDIM inversion process applied on first degraded frame. This promotes semantic consistency in the initial latent variables and enforces temporal consistency.

- **Synchronized Sampling Across Frames:** In VISION-XL, at each denoising timestep, the same noise schedule, prompt conditioning, and sampling hyper-parameters are used for all video frames. This synchronous scheduling guarantees that all frames evolve concurrently through the generative process, in spite of the fact that each frame is passed independently through the U-Net (with no inter-frame attention or weight sharing beyond the global network parameters). This simple approach can reduce temporal artifacts such as flickering, even though no explicit temporal priors or motion guidance are used.

- **Low-Pass Filtering:** As previously discussed, using a Gaussian blur kernel applied to the refined clean frame sequence suppresses high-frequency flickering artifacts in each frame and encourages smoother temporal changes.

- **Data Consistency via Conjugate Gradient Optimization:** CG optimization enforces data consistency across video frames. For spatio-temporal degradations including temporal blur, the optimization joins frames together through the degradation operator A, and enables information transfer between frames. For spatial degradations, temporal consistency is preserved through using shared settings to reduce flickering.

These mechanisms enable VISION-XL to achieve high-quality, temporally coherent video restoration results. However, the approach is still heuristic and lacks

explicit temporal priors or techniques. Several limitations of temporal mechanisms of VISION-XL are as follows:

- Shared latent initialization results in reduced motion sensitivity and can lead to over-smoothing in dynamic scenes.
- There is no explicit inter-frame communication in the sampling process.
- CG updates are applied independently and do not model temporal dependencies.

Unexplored Temporal Consistency Mechanisms. Beyond the inference-time mechanisms used in VISION-XL, several explicit temporal modeling techniques are used in the recent literature to improve coherence in video generation/restoration. A group of approaches are evolved around *video-based diffusion models*, in which the temporal consistency is embedded directly into the model architecture. For example, [Ho et al., 2022], extend 2D U-Nets to 3D by employing temporal convolutions or spatio-temporal attention layers. This architecture allows them to jointly process frame sequences. However, these methods require large-scale task-specific datasets, multi-frame training, and huge computational resources.

Another class of approaches utilize *motion-aware guidance*, where optical flow estimation or deformable convolution is used to align latent representations across frames [Bao et al., 2023]. Although this kind of techniques can enforce temporal consistency between frames, they may fail in the presence of occlusion, or inaccurate flow estimation in heavy degraded measurements. Furthermore, a lot of these approaches have limited modularity and generalization due to introducing significant architectural changes or extra training objectives.

These observations indicate that there is still space to explore lightweight, inference-time-compatible techniques in zero-shot image-based diffusion models domain for video restoration. Motivated by this gap, the subsequent three sections explore alternative approaches that can be integrated into the VISION-XL pipeline in zero-shot manner:

- **Cross-Frame Attention**: Enhancing temporal information exchange by sharing attention activations across neighboring frames.

- **Perceptual Straightening**: Straightening the restored video sequence in the space of human visual perception.

- **Sampling Trajectory Fusion**: Improving temporal consistency and reconstruction quality by fusing intermediate trajectories in diffusion sampling process.

These methods aim to preserve the zero-shot nature of VISION-XL while offer new foundations for temporally coherent video restoration.

4.3 Proposed Method: Cross-Frame Attention for Temporal Consistency

4.3.1 Background and Motivation

Several recent works have attempted to incorporate temporal awareness into diffusion frameworks by sharing attention across frames [Khachatryan et al., 2023]. In this way, it is possible to effectively reuse the learned spatial priors, enabling information exchange between frames. Their results suggest that these temporal cues can improve motion coherence and reduce perceptual flicker without retraining the diffusion backbone.

Inspired by these studies, we explore to see whether similar strategy can benefit zero-shot video restoration tasks. Particularly, we investigate whether attention sharing that implemented via cross-frame attention mechanisms can be integrated into the *VISION-XL* pipeline without modifying its diffusion prior or requiring additional temporal supervision.

4.3.2 Attention Mechanism Fundamentals

Mathematical Foundation of Attention. The attention mechanism calculates a weighted representation of input elements, where weights reflect the relevance of each element to a given query. Given a query vector $\mathbf{q} \in \mathbb{R}^{d_k}$ and key-value pairs $\{(\mathbf{k}_i, \mathbf{v}_i)\}_{i=1}^{n}$ where $\mathbf{k}_i \in \mathbb{R}^{d_k}$ and $\mathbf{v}_i \in \mathbb{R}^{d_v}$, attention computes:

$$\text{Attention}(\mathbf{q}, \mathbf{K}, \mathbf{V}) = \sum_{i=1}^{n} \alpha_i \mathbf{v}_i \tag{4.22}$$

where α_i are the attention weights which are computed by first calculating alignment scores between query and keys and then normalizing with softmax function:

$$e_i = f(\mathbf{q}, \mathbf{k}_i), \quad \alpha_i = \frac{\exp(e_i)}{\sum_{j=1}^{n} \exp(e_j)} \tag{4.23}$$

The most used compatibility function is scaled dot-product attention depicted in Figure 4.4:

$$f(\mathbf{q}, \mathbf{k}_i) = \frac{\mathbf{q}^T \mathbf{k}_i}{\sqrt{d_k}} \tag{4.24}$$

In matrix form we have:

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}}\right)\mathbf{V} \tag{4.25}$$

where $\mathbf{Q} \in \mathbb{R}^{m \times d_k}$, $\mathbf{K} \in \mathbb{R}^{n \times d_k}$, and $\mathbf{V} \in \mathbb{R}^{n \times d_v}$. The scaling factor $\frac{1}{\sqrt{d_k}}$ prevents gradient vanishing when d_k is large.

Cross-Attention Mechanism. Cross-attention allows one sequence/modality (image, text, ...) to attend to another and enables interaction between these two inputs. In cross-attention, queries are calculated from one input sequence and keys and values come from the other sequence. Mathematically, given two input sequences $\mathbf{X} \in \mathbb{R}^{m \times d_x}$ and $\mathbf{Y} \in \mathbb{R}^{n \times d_y}$, cross-attention computes queries from $\mathbf{X}$ and keys-values from $\mathbf{Y}$ using $\mathbf{W}_Q \in \mathbb{R}^{d_x \times d_k}$, $\mathbf{W}_K \in \mathbb{R}^{d_y \times d_k}$, and $\mathbf{W}_V \in \mathbb{R}^{d_y \times d_v}$ that are learned linear transformations:

$$\mathbf{Q} = \mathbf{X}\mathbf{W}_Q, \quad \mathbf{K} = \mathbf{Y}\mathbf{W}_K, \quad \mathbf{V} = \mathbf{Y}\mathbf{W}_V \tag{4.26}$$

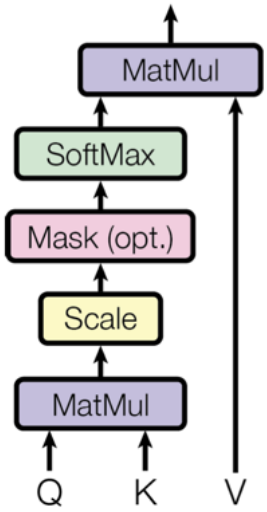

Figure 4.4: Scaled Dot-Product Attention [Vaswani et al., 2017]

The cross-attention is calculated then using the following formula, and the shape of the output is $m \times d_v$, where each position in $\mathbf{X}$ attends to all positions in $\mathbf{Y}$.

$$\text{CrossAttention}(\mathbf{X}, \mathbf{Y}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}}\right)\mathbf{V} \tag{4.27}$$

Cross-attention is used in multi-modal tasks for conditional generation. For example, in diffusion models such as Stable Diffusion [Rombach et al., 2022a] and SDXL, cross-attention allows spatial locations in the generated image to attend to relevant words in the text prompt and in this way conditions image generation on that prompt.

Self-Attention Mechanism. Self-attention has the same mechanism as cross-attention, but the difference is that queries, keys, and values are all computed from the same input sequence or in other words, different parts of the input token sequence attend to different parts of the same sequence. In spite of limited receptive field constraints of convolutional operations in CNNs, self-attention let the models capture long-range dependencies between spatial locations in images.

For an input image feature map $\mathbf{X} \in \mathbb{R}^{H \times W \times C}$, where H, W, and C represent height, width, and channels respectively, the feature map is first tokenized meaning

that reshaped into a sequence of spatial tokens $\mathbf{X}' \in \mathbb{R}^{n \times C}$ where $n = H \times W$.

The query, key, and value matrices are computed using learned linear transformations:

$$\mathbf{Q} = \mathbf{X}'\mathbf{W}_Q, \quad \mathbf{K} = \mathbf{X}'\mathbf{W}_K, \quad \mathbf{V} = \mathbf{X}'\mathbf{W}_V \tag{4.28}$$

where $\mathbf{W}_Q, \mathbf{W}_K \in \mathbb{R}^{C \times d_k}$ and $\mathbf{W}_V \in \mathbb{R}^{C \times d_v}$ are learnable parameter matrices.

Self-attention then computes:

$$\text{SelfAttention}(\mathbf{X}') = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}}\right)\mathbf{V} \tag{4.29}$$

Self-attention versus, cross-attention mechanism is illustrated in Figure 4.5

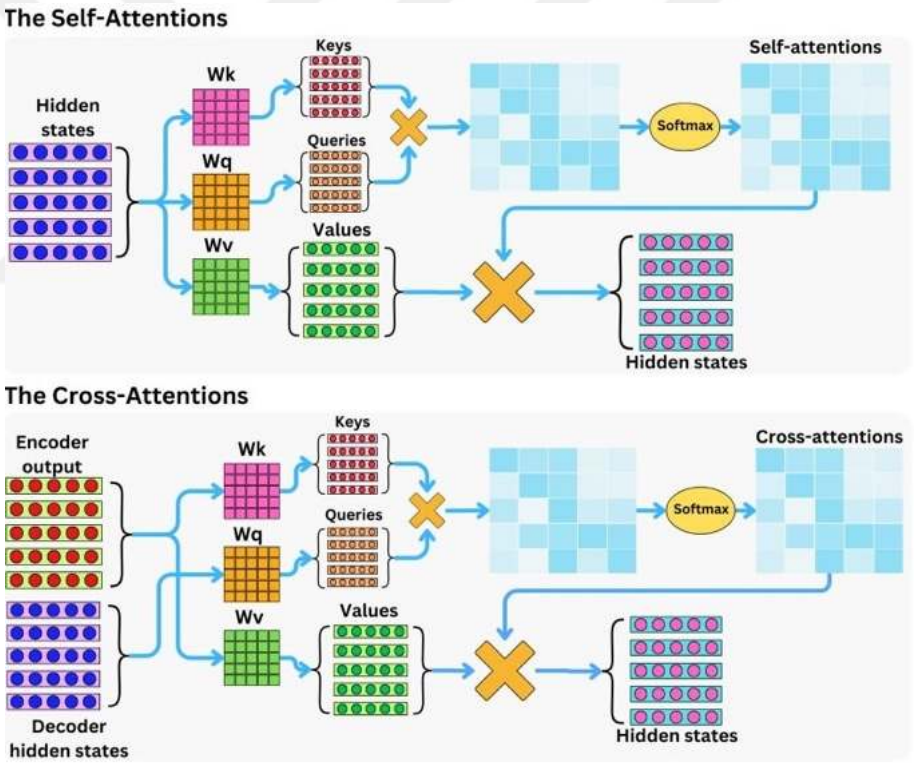

Figure 4.5: self attention vs. cross attention [Edge, 2025]

4.3.3 Attention in Diffusion Models

Generally, latent diffusion architectures integrate attention through transformer blocks embedded within the U-shaped encoder-decoder backbone called UNet. These

transformer blocks make long-range spatial interactions possible and can incorporate the conditioning information.

UNet Backbone and Transformer Blocks The multi-scale UNet architecture used in stable diffusion models including SDXL, consists of symmetric encoder and decoder with skip connections between corresponding resolution levels. Each stage in the UNet contains transformer blocks which use standard transformer module embedded within the residual block structure. These modules contain:

- **Self-Attention Layers**, which captures dependencies between spatial locations in the input feature map.
- **Cross-Attention Layers**, which are used for integrating the conditions, such as text embeddings or timestep embeddings and computes attention between input latent features and conditioning vectors.
- **Feedforward Networks (FFN)**, applied after attention layers to further process the attended features.
- **Normalization and Residual Connections**, including LayerNorm or GroupNorm, to stabilize training and preserve gradient flow.

Figure 4.6 illustrates general UNet structure and the transformer block.

Although transformer-based attention in diffusion models is effective for static conditional image generation, for video tasks it is limited to do self-attention on single image and can capture only intra-frame operations. In this work, based on this attention foundation, we will explore cross-frame attention strategies for enhancing temporal coherence in video restoration tasks, as detailed in the following sections.

4.3.4 Cross-Frame Attention for Video Processing

Limitations of Frame-Independent Processing Standard self-attention in diffusion models processes each frame independently, neglecting temporal correlations

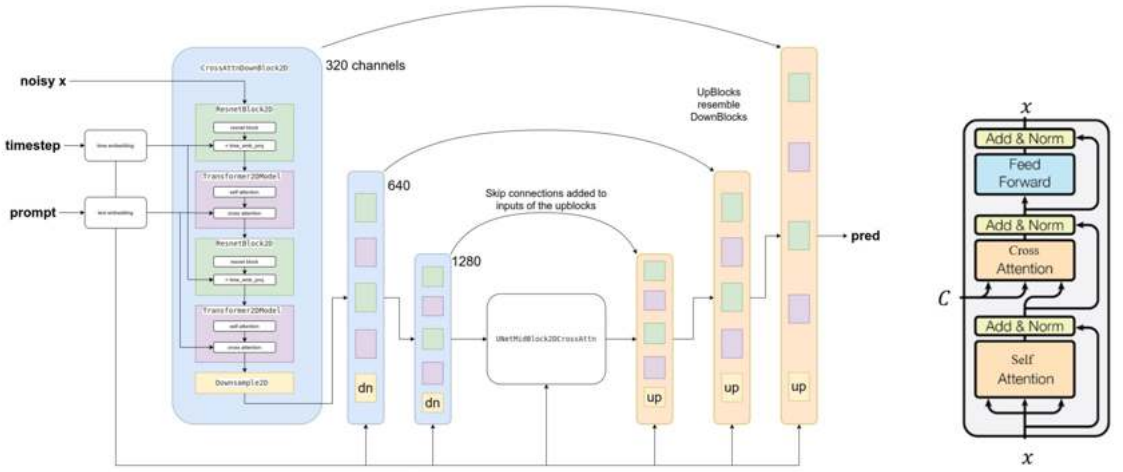

Figure 4.6: left:Schematic of the latent diffusion UNet architecture. right: transformer block structure within unet

essential for video consistency. For a given frame i, the attention query, key, and, value $\mathbf{Q}, \mathbf{K}, \mathbf{V}$ in Formula 4.29 are derived independently from frame i.

The key limitation is motion discontinuity resulting in jerky or flickering movement patterns.

Mathematical Formulation Cross-frame attention builds on standard attention, allowing the frames attend to the features from a set of reference frames [Khachatryan et al., 2023]. For frame i, we have:

$$\text{CrossFrameAttn}(\mathbf{Q}_i, \mathbf{K}_{\mathcal{R}(i)}, \mathbf{V}_{\mathcal{R}(i)}) = \text{softmax}\left(\frac{\mathbf{Q}_i \mathbf{K}_{\mathcal{R}(i)}^T}{\sqrt{d_k}}\right)\mathbf{V}_{\mathcal{R}(i)} \tag{4.30}$$

where $\mathcal{R}(i)$ represents a set of reference frames for frame i. The query $\mathbf{Q}_i$ is computed from the current frame, while keys and values $\mathbf{K}_{\mathcal{R}(i)}$ and $\mathbf{V}_{\mathcal{R}(i)}$ are calculated from reference frames, and in this way the attention weights can be interpreted as a measurement of temporal similarity and capture explicit temporal dependencies.

Integration with VISION-XL Pipeline In order to integrate the cross-frame attention mechanism, we define an attention processor function CrossFrameAttnProcessor and replace standard attention processors in selected UNet layers with this function.

Algorithm 1 Generalized Cross-Frame Attention Processor

Require: current frame index i, hidden state H_i, reference index set $S(\cdot)$, reference bank $\mathcal{R}$, projection functions $\textsc{ProjectQ}, \textsc{ProjectK}, \textsc{ProjectV}$, reference fusion $\textsc{FuseRef}$

Ensure: attention output O_i

$Q_i \leftarrow \textsc{ProjectQ}(H_i)$

$K_i \leftarrow \textsc{ProjectK}(H_i), \quad V_i \leftarrow \textsc{ProjectV}(H_i)$ ▷ Optionally store current frame features depending on strategy

$\mathcal{R} \leftarrow \textsc{UpdateRefBank}(\mathcal{R}, i, K_i, V_i, S)$ ▷ Ensure $\mathcal{R}$ contains key/value pairs for indices in $S(i)$

$(K_{\text{fused}}, V_{\text{fused}}) \leftarrow \textsc{FuseRef}(\mathcal{R}, S(i))$ ▷ $\textsc{FuseRef}$ may be identity, averaging, exponential blending, etc depending on the strategy.

$A_i \leftarrow \text{softmax}\Big(\frac{Q_i K_{\text{fused}}^{\top}}{\sqrt{d_k}}\Big)$

$O_i \leftarrow A_i V_{\text{fused}}$

return O_i

Algorithm 1 describes a generalized cross-frame attention mechanism that supports different temporal attention strategies in a frame-by-frame processing pipeline like Vision-XL. For the current frame i, the algorithm first computes the query projection Q_i from its hidden states. According to the chosen reference strategy $S(i)$, its key and value projections (K_i, V_i) are computed and stored into a reference bank $\mathcal{R}$, which is maintained so that it contains exactly the key/value pairs for the selected reference frames. These stored reference features are then fused by the $\textsc{FuseRef}$ function, which can be an arbitrary strategy for combining key/value pairs from one or multiple reference frames. The final attention output is calculated by computing attention weights between Q_i and the fused key K_{fused}, and applying them to the fused value V_{fused}.

4.3.5 Cross-Frame Attention Strategies

We modify the attention mechanism within all self-attention layers of the diffusion UNet and let the model attend not only to the features of the current frame but also to the adjacent or semantically relevant frames. This methodology potentially can capture inter-frame dependencies and help perceptual consistency across frames without modifying model weights or requiring additional training. To explore the effectiveness of this method, we implement and evaluate two different types of cross-frame attention strategies: (i) first frame reference and (ii) windowed multi-frame attention with exponential weighting. These strategies are described in detail below.

First Frame Reference Strategy Inspired by Text2Video-Zero methodology [Khachatryan et al., 2023], we integrated the first frame reference strategy in Vision-XL pipeline. This method forms a global temporal anchor by using the first frame as the reference for all subsequent frames in the sequence and provides a consistent reference point throughout the video generation process.

Mathematical Formulation: For any frame n, the reference set is defined as:

$$\mathcal{R}(n) = \{(K_1, V_1)\}, \quad \forall n \in \{1, 2, \ldots, N\} \tag{4.31}$$

This strategy results in the following attention pattern:

$$\mathbf{Y}_i = \begin{cases} \text{SelfAttn}(\mathbf{Q}_1, \mathbf{K}_1, \mathbf{V}_1) & \text{if } n = 1 \\ \text{CrossFrameAttention}(\mathbf{Q}_n, \mathbf{K}_1, \mathbf{V}_1) & \text{if } n > 1 \end{cases} \tag{4.32}$$

-Theoretical Advantages: Using the first frame as reference ensures that all frames share a common reference point and can provide strong consistency. Also, this method eliminates the error accumulation that can happen in sequential processing approaches. Through the shared key-value representations, the object structures in the first frame can propagated to all subsequent frames. Moreover, this strategy only requires storing the key-value pair of first frame, resulting in constant memory overhead independent of sequence length.

-Theoretical Limitations: Although this strategy is computationally efficient, as the video progresses, the first frame becomes less representative and this strategy

may not model scene changes, leading to temporal artifacts such as ghosting and flicker, especially in dynamic sequences. Without any warping or motion compensation, it may struggle to align features under large motion, resulting in attention mismatches.

Windowed Multi-Frame Attention with Weighting The next strategy we tried is windowed multi-frame attention strategy which combines the benefits of both previous approaches by allowing each frame to attend to multiple previous frames with exponentially decaying weights. This approach can balance temporal smoothness and (with a token budget) computational efficiency and mitigate error accumulation.

-Mathematical Formulation: For a given window size w and decay factor $\beta \in (0, 1)$, the reference set for frame i is:

$$\begin{aligned} \mathcal{S}(i) &= \{\max(0, i-w+1), \max(0, i-w+2), \ldots, i\} \\ \mathcal{R}(i) &= \{(\mathbf{K}_j, \mathbf{V}_j) \mid j \in \mathcal{S}(i)\} \end{aligned} \tag{4.33}$$

Instead of pre-softmax blending, we *concatenate* keys and apply exponential decay *only to values* from previous frames:

$$\begin{aligned} &\mathbf{K}_i^{\text{cat}} = \big[\mathbf{K}_j\big]_{j\in\mathcal{S}(i)}, \\ &\mathbf{V}_i^{\text{cat}} = \big[\,\gamma_{i,j}\,\mathbf{V}_j\,\big]_{j\in\mathcal{S}(i)}, \qquad \gamma_{i,j} = \begin{cases} 1, & j = i, \\ \beta^{\,i-j}, & j < i\ . \end{cases} \end{aligned} \tag{4.34}$$

$$\mathbf{Y}_i = \text{CrossFrameAttention}(\mathbf{Q}_i, \mathcal{R}(i)) = \text{softmax}\left(\frac{\mathbf{Q}_i \mathbf{K}_i^{\text{cat},T}}{\sqrt{d_k}}\right)\mathbf{V}_i^{\text{cat}}. \tag{4.35}$$

-Hyperparameter w: The window size w controls the temporal receptive field. Larger windows provide more context but increase computational cost.

-Hyperparameter β : The decay factor β controls the relative importance of distant frames. For $\beta \to 1$, all frames in the window receive similar weight on values, while $\beta \to 0$ approximates the original self-attention strategy.

-Theoretical Advantages: This strategy lets the model attend to previous frames, and the exponential decay reduces error accumulation by limiting the influence of distant frames. Moreover, the strategy can be adapted to different motion patterns through the hyper-parameters.

-Theoretical Limitations: Although this strategy is flexible, it may introduce several trade-offs. First, concatenation of multiple reference frames may still bias toward temporal averaging if w is large or decay is weak, especially with fast motion or scene transitions. Additionally, the method increases both computational and memory overhead compared to single-reference strategies (mitigated in practice via a token budget).

Conclusion. Cross-Frame Attention (CFA) injects temporal awareness into a pre-trained image LDM by letting each frame's queries attend to a fused bank of key–value features cached from reference frames. In this section we proposed the strategies to integrate cross-frame attention within VISION-XL to improve temporal coherence without retraining and with modest memory/compute and we discussed the potential advantages and limitations of each strategy.

4.4 Proposed Method: Perceptual Straightening Enhancement

4.4.1 Motivation

As discussed in Section 3.2.4, the perceptual straightening hypothesis (PSH), introduced by Hénaff et al. [Hénaff et al., 2019], states that the human visual system transforms naturally occurring video sequences to a perceptual feature space, where the videos follow straighter trajectories comparing to the pixel-intensity domain. This transformation reduces perceptual curvature and makes the progression of visual content more linear, and as a result, makes temporal prediction and understanding easier. **In contrast, unnatural or inconsistent visual sequences tend to exhibit greater curvature after this transformation**.

This neuroscientific hypothesis has inspired several works in video processing.

For example, Kancharla and Channappayya [Kancharla and Channappayya, 2021d] used PSH in a video frame prediction model and showed improvement in visual quality by minimizing trajectory curvature in the perceptual space. Moreover, based on this hypothesis, they proposed a blind metric in the context of no-reference quality assessment to evaluate the naturalness of motion in user-generated videos [Kancharla and Channappayya, 2021b].

In Section 3.3 and our paper [Rahimi and Tekalp, 2023], we extended this idea to video super-resolution (VSR), using perceptual straightness not as a constraint but as an evaluation metric.

Temporal consistency is still a challenge in diffusion-based video restoration, particularly in image-based zero-shot settings that there is no explicit motion modeling or supervision. In this work, we integrate perceptual straightness into the zero-shot VISION-XL pipeline and enforce straightness during the diffusion sampling. We hope this method can avoid unnatural or inconsistent restored sequences straightening their perceptual trajectory without explicit motion supervision.

4.4.2 Perceptual Straightening in VISION-XL

Although, VISION-XL framework achieves state-of-the-art performance in zero-shot video inverse problem solvers, it primarily focuses on data consistency and does not have an explicit mechanism to enforce temporal consistency. This is while, the human visual system exhibits remarkable sensitivity to temporal inconsistencies, particularly those that violate the principle of perceptual straightness across different levels of visual processing.

Our goal is to guide the diffusion denoising process toward restoring frames that follow straighter trajectories in the aforementioned perceptual representation space. Specifically, we propose to incorporate perceptual straightening as a refinement step during the VISION-XL sampling process. To achieve this, we need to define a perceptual straightening loss that penalizes video sequences exhibiting large curvature in the V1 representation and as a result, encouraging the generation of temporally

natural frames that align better with natural motion patterns as perceived by the human visual system.

Perceptual Representation Computation Given a sequence of N refined frames $\{\hat{x}_0^{[n]}\}_{n=1}^N$ obtained from the VISION-XL sampling process at timestep t, we compute hierarchical perceptual representations at different stages of the visual processing pipeline:

$$R_{\text{pixel}} = \{\hat{x}_0^{[n]}\}_{n=1}^N \tag{4.36}$$

$$R_{\text{retina}} = \{f_{\text{retina}}(\hat{x}_0^{[n]}\}_{n=1}^N \tag{4.37}$$

$$R_{\text{V1}} = \{f_{\text{V1}}(f_{\text{retina}}(\hat{x}_0^{[n]}\}_{n=1}^N \tag{4.38}$$

where $f_{\text{retina}}(\cdot)$ represents the RetinalDN transformation and $f_{\text{V1}}(\cdot)$ denotes the steerable pyramid processing that simulates V1 complex cell responses.

As illustrated in Figure 4.7, each representation forms a trajectory in high-dimensional space, where temporal consistency is measured through trajectory curvature.

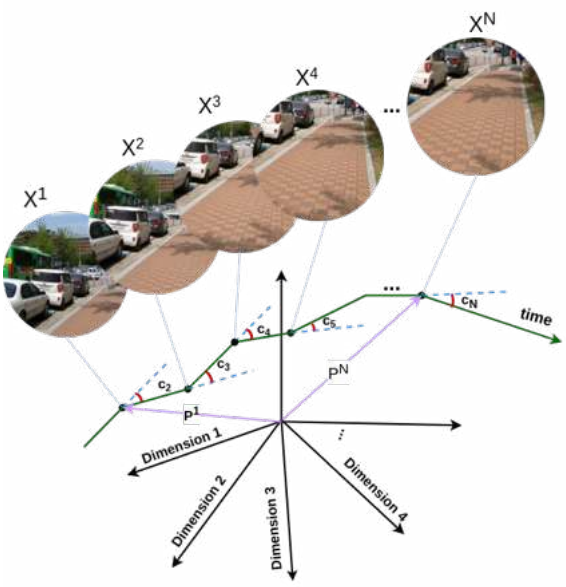

Figure 4.7: Visualization of a high-dimensional perceptual representation of a video sequence and the concept of curvature

Curvature Computation For each representation $R \in \{R_{\text{pixel}}, R_{\text{V1}}\}$, the curvature at frame n is computed as the angle between successive displacement vectors. Given perceptual embeddings $P^n \in R$ for frame n, we define displacement vectors:

$$V^n = P^n - P^{n-1}, \quad n = 2, 3, ..., N \tag{4.39}$$

The curvature at frame n is then calculated as:

$$\text{Curv}(n) = \arccos\left(\frac{V^n \cdot V^{n+1}}{\|V^n\|\|V^{n+1}\|}\right) \tag{4.40}$$

where the dot product measures the similarity between consecutive displacement directions. To ensure numerical stability, we incorporate a small epsilon value $\epsilon = 10^{-8}$ when computing vector magnitudes:

$$d = \sqrt{\sum(V^2) + \epsilon} \tag{4.41}$$

The overall trajectory curvature is obtained by averaging across the sequence:

$$\text{Curv}(R) = \frac{1}{N-2}\sum_{n=2}^{N-1}\text{Curv}(n) \tag{4.42}$$

This high-dimensional curvature computation detect subtle temporal inconsistencies that may not be visible in pixel space and provides a more perceptually meaningful measure of temporal naturalness.

Loss Formulation We formulate the perceptual straightness loss $\mathcal{L}_{PS}$ as a differentiable objective that can be seamlessly integrated into the VISION-XL optimization pipeline. Given a sequence of predicted clean frames $\hat{X}_0 = \{\hat{x}_0^{[n]}\}_{n=1}^N$ at timestep t during the reverse diffusion process, the loss is defined as:

$$\mathcal{L}_{\text{PS}}(\hat{X}_0) = \text{ReLU}(\text{Curv}_{V1}(\hat{X}_0)) \tag{4.43}$$

This formulation is motivated by several key considerations. First, the goal is to penalize video frames with large V1 curvature, thereby encouraging restoring the sequences that follow smoother trajectories in the perceptual space.

Second, the ReLU activation function is used primarily for numerical stability and to ensure non-negative loss values, though since curvature values are inherently

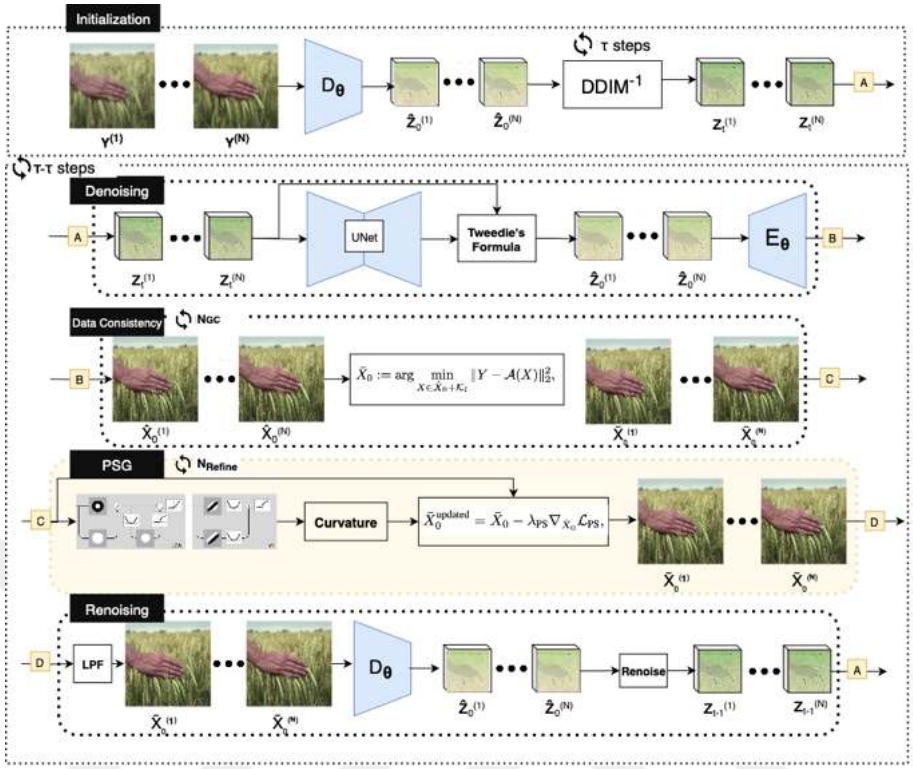

Figure 4.8: The proposed PSG block within inference process of VISION-XL.

non-negative (angles between consecutive displacement vectors range from 0° to 180°), its impact is minimal.

4.4.3 Perceptual Straightening Optimization Strategies

We incorporate the perceptual straightening penalty as guidance during diffusion sampling. As illustrated in Figure 4.8, at each denoising timestep after predicting the clean sequence, we iteratively compute $\mathcal{L}_{\text{PS}}$ and back-propagate the penalty computed in perceptual space to refine the predicted clean frames using gradient descent:

$$\bar{X}_0^{\text{updated}} = \bar{X}_0 - \lambda_{\text{PS}} \nabla_{\bar{X}_0} \mathcal{L}_{\text{PS}}, \tag{4.44}$$

where λ_{PS} is a tunable guidance weight, controlling the influence of the perceptual straightening penalty during sampling.

The details of PSG process is described in Algorithm 2.

This regularization steers the model toward generating smoother, temporally aligned outputs without requiring motion supervision or reference video data.

Algorithm 2 Perceptual Straightness Refinement (PSRefinement)

Require: Input frames $\mathbf{X}_{\text{in}}$, number of iterations N_{refine}

Ensure: Perceptually refined frames $\tilde{\mathbf{X}}_0$

$\tilde{\mathbf{X}}_0 \leftarrow \mathbf{X}_{\text{in}}$.requires_grad_(True)

for $i = 1$ **to** N_{refine} **do**

  $\mathcal{L}_{PS} \leftarrow \text{PerceptualStraightnessLoss}(\tilde{\mathbf{X}}_0)$

  $\nabla \mathcal{L}_{PS} \leftarrow \text{autograd.grad}(\mathcal{L}_{PS}, \tilde{\mathbf{X}}_0)$

  $\tilde{\mathbf{X}}_0 \leftarrow \tilde{\mathbf{X}}_0 - \lambda_{\text{PS}} \cdot \nabla \mathcal{L}_{PS}$

  if $(i + 1) \bmod 2 = 0$ **then**

    $\lambda_{\text{PS}} \leftarrow \lambda_{\text{PS}} \times 0.8$ ▷ Adaptive learning rate

return $\tilde{\mathbf{X}}_0$

To avoid the overhead of decoding latent sequences into pixel-space repeatedly, we decided to integrate the perceptual straightness loss during the pixel-space optimization phase of the VISION-XL pipeline. We explored three different strategies to effectively combine data consistency refinement (DDS) with perceptual straightening optimization. Each of these strategies offer different trade-offs between computational efficiency and optimization effectiveness.

Strategy 1: Sequential Optimization In the first approach we apply data consistency refinement followed by perceptual straightening optimization in separate sequential phases:

Algorithm 3 Sequential DDS + PS Optimization

Require: Initial predicted clean frames $\hat{\mathbf{X}}_0$ from denoising timestep t, measurement frames $\mathbf{Y}$

Ensure: Optimized frames $\tilde{\mathbf{X}}_0$

$\bar{\mathbf{X}}_0 \leftarrow \text{DDS}(\hat{\mathbf{X}}_0, \mathbf{Y}, N_{\text{DDS}})$ ▷ Data consistency via CG

$\tilde{\mathbf{X}}_0 \leftarrow \text{PSRefinement}(\bar{\mathbf{X}}_0, N_{\text{PS}})$ ▷ Perceptual refinement

return $\tilde{\mathbf{X}}_0$

This sequential approach optimizes each component with its optimal solver: conjugate gradient for the quadratic data fidelity term (DDS) and gradient descent for the non-convex perceptual straightening objective.

Strategy 2: Alternating Optimization As the second strategy, we alternate between single iterations of data consistency and perceptual straightening loss:

Algorithm 4 Alternating DDS + PS Optimization

Require: Initial predicted clean frames $\hat{\mathbf{X}}_0$ from denoising timestep t, measurement frames $\mathbf{Y}$

Ensure: Optimized frames $\tilde{\mathbf{X}}_0$

$\tilde{\mathbf{X}}_0 \leftarrow \hat{\mathbf{X}}_0$

for $i = 1$ **to** N_{alt} **do**

  $\tilde{\mathbf{X}}_0 \leftarrow \text{DDS}(\tilde{\mathbf{X}}_0, \mathbf{Y}, 1)$ ▷ Single DDS iteration

  $\tilde{\mathbf{X}}_0 \leftarrow \text{PSRefinement}(\tilde{\mathbf{X}}_0, 1)$ ▷ Single PS iteration

return $\tilde{\mathbf{X}}_0$

This approach can enable better interaction between data fidelity and perceptual objectives throughout the optimization process and may lead to better convergence point.

Strategy 3: Joint Optimization In the third approach, we combine both objectives into a unified loss function to optimize it simultaneously:

Algorithm 5 Joint DDS + PS Optimization

Require: Initial predicted clean frames $\hat{\mathbf{X}}_0$ from denoising timestep t, measurement frames $\mathbf{Y}$

Ensure: Optimized frames $\tilde{\mathbf{X}}_0$

$\tilde{\mathbf{X}}_0 \leftarrow \hat{\mathbf{X}}_0$.requires_grad_(True)

Initialize Adam optimizer with $\tilde{\mathbf{X}}_0$

for $i = 1$ **to** N_{joint} **do**

  $\mathcal{L}_{\text{fidelity}} \leftarrow \|\mathbf{Y} - A(\tilde{\mathbf{X}}_0)\|_2^2$

  $\mathcal{L}_{PS} \leftarrow \text{PerceptualStraightnessLoss}(\tilde{\mathbf{X}}_0)$

  $\mathcal{L}_{\text{total}} \leftarrow \mathcal{L}_{\text{fidelity}} + \lambda_{\text{PS}} \cdot \mathcal{L}_{PS}$

  Update $\tilde{\mathbf{X}}_0$ using Adam optimizer on $\mathcal{L}_{\text{total}}$

return $\tilde{\mathbf{X}}_0$

This joint optimization approach needs careful balancing of loss scales (λ_{PS}), as the data fidelity loss operates on a significantly different magnitude than the perceptual straightening loss.

The comparative analysis and empirical evaluation of these strategies are presented in the Results section.

Conclusion. In this section we formalized perceptual straightening in VISION-XL by mapping predicted clean frames into the retinal–V1 feature space and refining the predicted clean video sequence to have straighter trajectory in perceptual space, In this way we can steer sampling toward straighter, more natural and predictable motion.

4.5 Proposed Method: Multi-Path Ensemble Sampling for Robust Solution

4.5.1 Introduction and Motivation

In diffusion-based video restoration models, such as Vision-XL [Kwon and Ye, 2024], the intrinsic stochastic nature of diffusion models causes inevitable variation in reconstructed outputs. During the denoising process, from the noise toward the data distribution, at each denoising step, random noise is injected to guide the sampling trajectory [Ho et al., 2020, Song et al., 2020]. Although this stochasticity is crucial in generative models, it can cause problems especially in inverse problems, where we want to come up with a single, optimal reconstruction.

Each time a diffusion model, such as Vision-XL, processes the same input, the noisy latent follows a slightly different trajectory because of the randomness in the denoising process. This trajectory variability can cause small differences in pixel values and subsequently in fine-grained details, as well as inconsistencies in temporal dynamics of the video frames [Kwon and Ye, 2024, Daras et al., 2024]. The iterative nature of diffusion sampling can make the situation worse, where small errors and noise are accumulated at each denoising step.

To mitigate this problem, we propose an ensembling approach, which is known to reduce the prediction variance. The main idea in ensembling approaches is to combine multiple independent predictions [Dietterich, 2000]. Although individual predictions may be noisy, their average is likely to be more accurate. Statistically, it is also known that averaging multiple independent estimations leads to better approximation of the ground truth [Efron, 2011, Bishop and Nasrabadi, 2006].

These observations inspired us to investigate the multi-path ensemble sampling for video inverse problems. Instead of relying on a single diffusion trajectory, we suggest generating multiple sampling paths and intelligently fusing these trajectories to achieve potentially better results.

4.5.2 Sources of Stochasticity in Diffusion-Based Video Restoration

As discussed earlier, the application of diffusion-based models to inverse problems introduces several sources of stochasticity that can affect reconstruction quality and temporal consistency. Understanding these sources is essential for developing effective ensemble strategies.

Random Noise Injection During Renoising The main source of randomness in VISION-XL pipeline comes from the random noise injection in the re-noising step during the inference. As discussed in 4.2.4, at the end of each denoising iteration, VISION-XL injects a random noise to the latent for the next timestep:

$$\begin{aligned} c_1 &= \sqrt{1-\bar{\alpha}_{t-1}} \cdot \eta \\ c_2 &= \sqrt{1-\bar{\alpha}_{t-1}} \cdot \sqrt{1-\eta^2} \\ \mathbf{z}_{t-1} &= \sqrt{\bar{\alpha}_{t-1}}\bar{\mathbf{z}}_0 + c_1 \cdot \boldsymbol{\epsilon}_{\text{fixed}} + c_2 \cdot \boldsymbol{\epsilon}_\theta(\mathbf{z}_t, t) \end{aligned} \tag{4.45}$$

At each denoising step, the random noise $\boldsymbol{\epsilon}_{fixed} \sim \mathcal{N}(0, \mathbf{I})$ is fixed across all frames of the video but is sampled independently for different time steps. This randomness allows different sampling paths to diverge even with identical initial conditions.

Score Function Uncertainty Due to the difference in injected noise, the denoising paths begin to diverge and therefore the predicted noise by UNet becomes different for the subsequent timesteps. While the pretrained U-Net is deterministic for identical inputs, different latent states $\mathbf{z}_t^{(1)} \neq \mathbf{z}_t^{(2)}$ lead to different noise predictions:

$$\boldsymbol{\epsilon}_\theta(\mathbf{z}_t^{(1)}, t) \neq \boldsymbol{\epsilon}_\theta(\mathbf{z}_t^{(2)}, t) \tag{4.46}$$

This divergence causes an extra stochasticity in the denoising trajectory, as multiple valid paths exist in the learned score field.

VAE Encoding and Decoding Artifacts Built based on SDXL, VISION-XL works in the latent space. The pre-trained VAE, introduces different reconstruction errors for slightly different inputs which are particularly problematic during

iterative sampling. Repetitive encode-decodes can drift the latent from the clean manifold $\mathcal{M}_0$.

Data Consistency Optimization Variability As discussed in 4.2.4 VISION-XL uses conjugate gradient optimization to enforce data fidelity with the measurement:

$$\bar{\mathbf{x}}_0 := \arg \min_{X \in \hat{X}_0 + \mathcal{K}_l} \|Y - A(X)\|_2^2 \tag{4.47}$$

This can be another source of randomness due to the fact that different predicted clean images $\hat{X}_0$ can lead to different optimization trajectories and local minimums.

The initial differences caused by noise injection, propagate and amplify through the sampling trajectory and lead to significantly different final reconstructions. The key insight is that while the pretrained UNet itself is deterministic, the different injected noise in different parallel reconstructions leads to multiple valid sampling trajectories. These trajectories can explore different regions of the data distribution through diffusion prior. Therefore, ensembling strategies can utilize this diversity to achieve more robust and consistent reconstructions.

4.5.3 Multi-Path Ensemble Sampling

We propose a multi-path ensemble sampling framework that generates multiple synchronized diffusion trajectories and fuses them to obtain more robust video reconstructions. The key idea is to exploit the statistical advantages of ensemble methods while preserving the temporal consistency of video restoration results.

General Framework Let $\mathcal{S}_\theta$ express a diffusion-based inverse problem solver (such as VISION-XL) that takes degraded measurements $\mathbf{Y}$ as input and generates a restored video sequence $\mathbf{X}$. Our multi-path ensemble approach produces K independent sampling trajectories:

$$\mathbf{X}^{(k)} = \mathcal{S}_\theta(\mathbf{Y}, \boldsymbol{\xi}^{(k)}), \quad k = 1, 2, \ldots, K \tag{4.48}$$

where $\boldsymbol{\xi}^{(k)}$ denotes the random seed or stochastic components unique to the k-th path. We use a fusion function $\mathcal{F}$ to obtain the final ensemble prediction:

$$\mathbf{X}_{ensemble} = \mathcal{F}(\mathbf{X}^{(1)}, \mathbf{X}^{(2)}, \ldots, \mathbf{X}^{(K)}) \tag{4.49}$$

Independent Multi-Path Sampling One of the most direct strategies for multi-path ensemble sampling is to run K independent diffusion processes and fuse their outputs at the end. In this approach, each path can evolve independently throughout the sampling process:

Algorithm 6 Independent Multi-Path Sampling

Require: Measurements $\mathbf{Y}$, Number of paths K

for $k = 1, \ldots, K$ **do**
  $\mathbf{z}_\tau^{(k)} \leftarrow \text{BatchConsistentInversion}(\mathbf{Y}, \boldsymbol{\xi}^{(k)})$
  for $t = \tau, \tau-1, \ldots, 1$ **do**
    $\hat{\mathbf{z}}_0^{(k)} \leftarrow \text{TweedieDenoising}(\mathbf{z}_t^{(k)}, t)$ (Formula 4.17)
    $\hat{\mathbf{X}}_0^{(k)} \leftarrow \text{VAEDecode}(\hat{\mathbf{z}}_0^{(k)})$
    $\bar{\mathbf{X}}_0^{(k)} \leftarrow \text{DataConsistency}(\hat{\mathbf{X}}_0^{(k)}, \mathbf{Y})$ (Formual 4.18)
    $\mathbf{z}_{t-1}^{(k)} \leftarrow \text{Renoise\&Encode}(\bar{\mathbf{X}}_0^{(k)}, \boldsymbol{\epsilon}_t^{(k)})$ (Formula 4.20)
return $\mathcal{F}_{final}(\bar{\mathbf{X}}_0^{(1)}, \ldots, \bar{\mathbf{X}}_0^{(K)})$

As clarified in Algorithm 6, in this method, each sampling trajectory $\{\mathbf{z}_t^{(k)}\}_{t=0}^{\tau}$ creates its own independent path through the latent manifold. As a result, the paths can diverge significantly during the sampling process and may explore different regions of the posterior distribution $p(\mathbf{X}|\mathbf{Y})$ (Figure 4.9). The final ensemble prediction is obtained using proper fusion like simple uniform averaging.

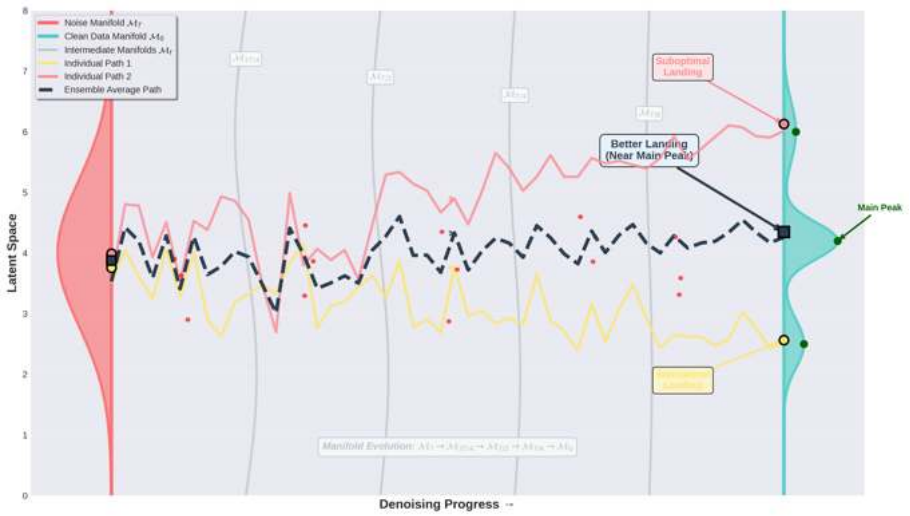

Figure 4.9: Multi-Path Ensemble Sampling: Individual vs. Average Trajectories. Two Paths Land sub-optimally, Ensemble Average Finds Better Solution

Synchronized Multi-Path Sampling As the second approach, we suggest a synchronized sampling scheme where different paths interact and share information at each denoising timestep. For timestep t, let $\mathbf{z}_t^{(k)}$ show the latent state of the k-th path. This approach is explained by Algorithm 7 and the evolution of trajectories are illustrated in Figure 4.10 for better understanding.

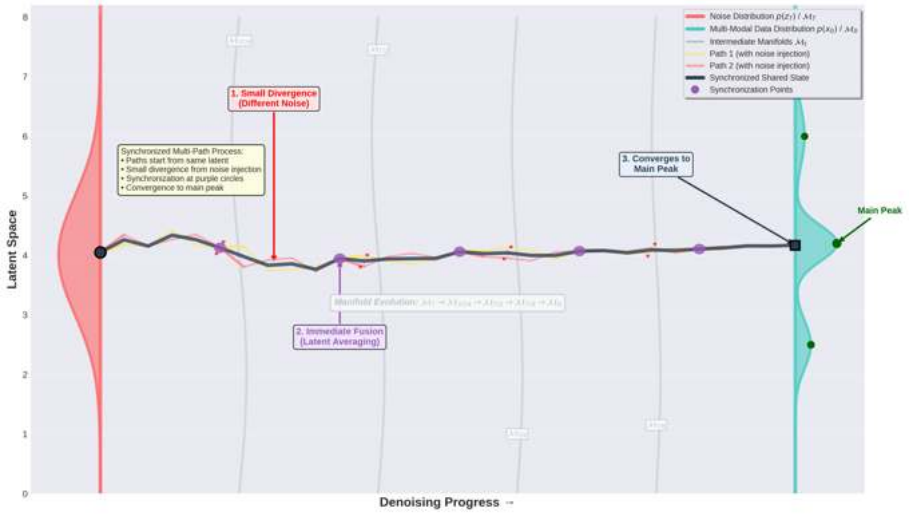

Figure 4.10: Synchronized Multi-Path Sampling. Individual vs. Average Trajectories.

Algorithm 7 Multi-Path Synchronized Sampling

Require: Measurements $\mathbf{Y}$, Number of paths K, Fusion weights $\mathbf{w}$

Initialize $\mathbf{z}_t^{(k)} \leftarrow \text{BatchConsistentInversion}(\mathbf{Y}, \boldsymbol{\xi}^{(k)})$ for $k = 1, \ldots, K$

for $t = \tau, \tau - 1, \ldots, 1$ **do**

  for $k = 1, \ldots, K$ **do**

    $\hat{\mathbf{z}}_0^{(k)} \leftarrow \text{TweedieDenoising}(\mathbf{z}_t^{(k)}, t)$ (Formula 4.17)

    $\hat{\mathbf{X}}_0^{(k)} \leftarrow \text{VAEDecode}(\hat{\mathbf{z}}_0^{(k)})$

    $\bar{\mathbf{X}}_0^{(k)} \leftarrow \text{DataConsistency}(\hat{\mathbf{X}}_0^{(k)}, \mathbf{Y})$ (Formual 4.18)

    $\mathbf{z}_{t-1}^{(k)} \leftarrow \text{Renoise\&Encode}(\bar{\mathbf{X}}_0^{(k)}, \boldsymbol{\epsilon}_t^{(k)})$ (Formula 4.20)

  $\mathbf{z}_{fused} \leftarrow \mathcal{F}_{latent}(\mathbf{z}_{t-1}^{(1)}, \ldots, \mathbf{z}_{t-1}^{(K)}, \mathbf{w})$

  $\mathbf{z}_{t-1}^{(k)} \leftarrow \mathbf{z}_{fused}$ for all k ▷ Synchronization

return $\mathcal{F}_{final}(\bar{\mathbf{X}}_0^{(1)}, \ldots, \bar{\mathbf{X}}_0^{(K)})$

Fusion Strategies Ensemble fusion can be done using different weighting schemes and at different representation spaces. we can consider following fusion approaches :

Weighting Strategies:

- **Uniform Averaging:**

$$\mathcal{F}_{uniform}(\mathbf{R}^{(1)}, \ldots, \mathbf{R}^{(K)}) = \frac{1}{K} \sum_{k=1}^{K} \mathbf{R}^{(k)} \tag{4.50}$$

- **Weighted Averaging:**

$$\mathcal{F}_{weighted}(\mathbf{R}^{(1)}, \ldots, \mathbf{R}^{(K)}) = \sum_{k=1}^{K} w_k \mathbf{R}^{(k)}, \quad \sum_{k=1}^{K} w_k = 1 \tag{4.51}$$

- **Quality-Aware Fusion:**

$$w_k = \frac{\exp(-\lambda \mathcal{L}_{data}(\mathbf{X}^{(k)}, \mathbf{Y}))}{\sum_{j=1}^{K} \exp(-\lambda \mathcal{L}_{data}(\mathbf{X}^{(j)}, \mathbf{Y}))} \tag{4.52}$$

  where $\mathcal{L}_{data}(\mathbf{X}^{(k)}, \mathbf{Y}) = \|\mathbf{Y} - \mathcal{A}(\mathbf{X}^{(k)})\|_2^2$ measures the data consistency of the k-th path and encourages the fused trajectory toward the one with better data consistency.

Representation Spaces:

$\mathbf{R}^{(k)}$ in the weighting strategies can denote either latents or decoded images depending on the representation spaces we choose to do fusion:

- **Latent-Space Fusion:**

$$\mathbf{Z}_{ensemble} = \mathcal{F}(\mathbf{Z}_0^{(1)}, \mathbf{Z}_0^{(2)}, \ldots, \mathbf{Z}_0^{(K)}) \tag{4.53}$$

  followed by $\mathbf{X}_{ensemble} = \mathcal{D}_\theta(\mathbf{Z}_{ensemble})$

- **Pixel-Space Fusion:**

$$\mathbf{X}_{ensemble} = \mathcal{F}(\mathbf{X}_0^{(1)}, \mathbf{X}_0^{(2)}, \ldots, \mathbf{X}_0^{(K)}) \tag{4.54}$$

  where $\mathbf{X}_0^{(k)} = \mathcal{D}_\theta(\mathbf{Z}_0^{(k)})$ represents the decoded images from each path.

4.5.4 Theoretical Analysis

The idea of ensembling multiple diffusion trajectories that we use in video restoration is supported by different theoretical point views.

Manifold Regularization via Trajectory Averaging As illustrated in Figures 4.9 and 4.10, diffusion sampling can be considered as a trajectory across a sequence of latent manifolds $\mathcal{M}_t$, starting from pure noise $\mathcal{M}_T$ and moving toward the clean data manifold $\mathcal{M}_0$. Each sample path $\{\mathbf{z}_t^{(k)}\}_{t=0}^T$ may deviate from the related manifold because of the accumulated noise or optimization errors.

Averaging latent from multiple trajectories at each timestep (manifold) has a regularizing effect: when individual latents are near the manifold, their Euclidean average will also stay near the manifold.

Posterior Smoothing in MMSE Estimation As discussed earlier in 4.2.3 diffusion-based inverse solvers approximate sampling from the posterior distribution

$p(\mathbf{X}|\mathbf{Y})$. Under squared loss, the optimal Bayesian estimator called the Minimum Mean Squared Error (MMSE) estimator is the posterior mean:

$$\mathbf{X}_{MMSE} = \mathbb{E}[\mathbf{X}|\mathbf{Y}] = \int \mathbf{X}\, p(\mathbf{X}|\mathbf{Y})\, d\mathbf{X} \tag{4.55}$$

This estimator achieves the reconstruction with the minimum expected squared difference from the ground truth. For high-dimensional distributions that the exact estimation of this expectation is not possible, it can be approximated using Monte Carlo sampling [Robert and Casella, 2004]:

$$\mathbf{X}_{MMSE} \approx \frac{1}{K} \sum_{k=1}^{K} \mathbf{X}^{(k)}, \quad \mathbf{X}^{(k)} \sim p(\mathbf{X}|\mathbf{Y}) \tag{4.56}$$

According to the central limit theorem, the approximation error which is the standard deviation of $\mathbf{X}^{(k)}$ decreases as $O(1/\sqrt{K})$. These theoretical analysis offers a foundation for understanding why multi-path ensemble sampling potentially can improve the quality of restored video.

Conclusion. In this section we introduced a multi-path ensemble framework that runs K diffusion trajectories and combines them with different strategies (synchronized or independently, in latent or pixel space) to yield a single, more robust reconstruction. We also showed theoretically that averaging along trajectories regularizes proximity to the data manifold and uses stochastic path diversity to reduce variance. This provides a principled, training-free robustness boost for video inverse problems and motivates the empirical ablation reported in next section.

4.6 Evaluation

4.6.1 Evaluation Setup

Datasets Similar to the baseline (VISION-XL), we evaluate our developed methods on the DAVIS dataset. Moreover, to maintain consistency with the evaluation dataset used for the specialized super-resolution models studied in Chapter 3, we also include the REDS4 dataset:

-DAVIS Dataset We use 73 sequences of 25 frames each from the DAVIS-2017 dataset [Pont-Tuset et al., 2017]. We resized each frame to 768×1280 resolution following the protocol used by VISION-XL [Kwon and Ye, 2024]. The selected sequences maintain their original landscape orientation. The DAVIS dataset includes high-quality video sequences with challenging scenarios and diverse temporal dynamics making it ideal for video restoration tasks and spatio-temporal analysis.

-REDS4 Dataset As the second test dataset, we incorporate four sequences (clips 000, 011, 015 and 020 - called REDS4) from REalistic and Diverse Scenes (REDS) dataset [Nah et al., 2019], which contains high-resolution video sequences with large and complex motions. Each video sequence contained 100 frames in landscape orientation with resolution of 720×1280. Containing different motion patterns and scene complexities compared to DAVIS, REDS4 provides different evaluation scenarios to evaluate the generalization ability of the models.

Degradation Models We experimented with a wide range of spatio-temporal inverse problems using seven different degradations. Each degradation is formulated as a composition of temporal and spatial operators, following the framework established in VISION-XL [Kwon and Ye, 2024]:

Spatial-Only Degradations

1. **Super-Resolution (SR)**: To degrade the frames 4× spatial downsampling is applied using adaptive average pooling.

2. **Deblurring**: To blur the frames Gaussian spatial blur with kernel size 61×61 and $\sigma = 3.0$ is applied with reflection padding.

3. **Inpainting**: For inpainting tasks, random binary masking is applied with 50% pixel occlusion ratio.

Spatio-Temporal Degradations

4. **Temporal Blur**: As temporal degradation we employ 13-frame temporal averaging using uniform blur kernel with circular padding

5. **SR+ (SR + Temporal Blur)**: We apply combination of 4× spatial downsampling and 7-frame temporal averaging using uniform kernel

6. **Deblur+ (Deblur + Temporal Blur)**: We use combination of Gaussian spatial blur and 7-frame temporal averaging using uniform kernel

7. **Inpaint+ (Inpaint + Temporal Blur)**: We use combination of random binary masking and 7-frame temporal averaging using uniform kernel

The measurement model is defined as $\mathbf{y} = \mathcal{A}(\mathbf{x})$, in which $\mathcal{A}$ is the degradation operator and $\mathbf{x}$ is the ground truth video sequence.

Evaluation Metrics To evaluate both perceptual quality and fidelity of the result sequences, we adopt a variety of metrics discussed in Section 2.5.

-Spatial Fidelity Metrics

- Peak Signal-to-Noise Ratio (PSNR) (dB)
- Structural Similarity Index (SSIM)
- Mean Squared Error (MSE)

-Perceptual Quality Metrics

- Learned Perceptual Image Patch Similarity (LPIPS)
- Deep Image Structure and Texture Similarity (DISTS)
- Edge-based Reference-less Quality Assessment (ERQA)
- Natural Image Quality Evaluator (NIQE)

-Temporal Metrics

- Optical Flow MSE (OF-MSE)
- Fréchet Video Distance (FVD)
- Temporal Profile Analysis
- Perceptual Straightness (PS)
- Spatio-Temporal Perceptual Measure (P-ST)
- Spatio-Temporal Distortion Measure (D-ST)

In order to be consistent with the evaluation setup of Chapter 3, along with OF-MSE and FVD, we employ Perceptual straightness/Curvature discussed earlier in both Chapters 3 and 4. Moreover, we use new spatio-temporal metrics, including Spatio-Temporal Perceptual Measure (P-ST) and Spatio-Temporal Distortion Measure (D-ST) introduced in subsections 3.3.1 and 3.3.2 respectively.

Implementation Details All experiments are conducted on a single NVIDIA V100 GPU with 32GB memory. Also, all of the inferences are done with mixed precision computation using half precision (torch.float16) for memory efficiency.

4.6.2 Baseline Performance: VISION-XL

As our primary baseline we use VISION-XL [Kwon and Ye, 2024] which is the current state-of-the-art in diffusion-based zero-shot video inverse problem solver. VISION-XL utilizes the text-to-image latent-based Stable Diffusion XL 1.0 (SDXL) as its backbone along with its enhanced VAE (madebyollin/sdxl-vae-fp16-fix). This framework employs a DDIM scheduler with 25 sampling steps (NFE=25).

The framework is discussed in detail in Section 4.2.4. To manage memory constraints, the method implements pseudo-batch consistent sampling, which processes the frames individually while maintaining temporal consistency using batch-consistent inversion initialization. They apply classifier-free guidance using null-text

conditioning and with a scale factor of 0.6 to minimize the unwanted guidance from text embeddings.

The hyper-parameters are set according to the optimal values reported in their paper: inversion time $\tau = 0.3T$ (8 timesteps) , low-pass filtering scale $\lambda = 2$, conjugate gradient steps $l = 10$ for data-driven sampling (DDS), and DDIM eta parameter $\eta = 0.2$ for controlled stochasticity in renoising step.

We performed two experiments with same hyper-parameters but different latent initialization approaches. The **Baseline** experiment is according to the original VISION-XL protocol, in which they used batch-consistent latent initialization doing DDIM inversion on the first measurement frame and then replicating this latent across all frames. In the **Baseline+** experiment, we initialize each frame with DDIM inversion of its corresponding measurement frame resulting in frame-specific latent initialization. This modification provides more frame-specific information during initialization and we want to assess whether individualized initialization can improve reconstruction quality at the cost of increased runtime overhead.

Representative reconstruction results for the Baseline are presented for the DAVIS and REDS4 datasets across different tasks, including SR/SR+ (DAVIS: Figure 4.11, REDS4: Figure 4.14), Deblur/+Deblur (DAVIS: Figure 4.12, REDS4: Figure 4.15), and Inpaint/+Inpainting (DAVIS: Figure 4.13, REDS4: Figure 4.16

Figure 4.11: Representative comparison of two sequential frames from **DAVIS** dataset, **Left: SR, right: SR+** tasks. From top to bottom: measurement, reconstructed by **Baseline** (VISION-XL), and ground truth.

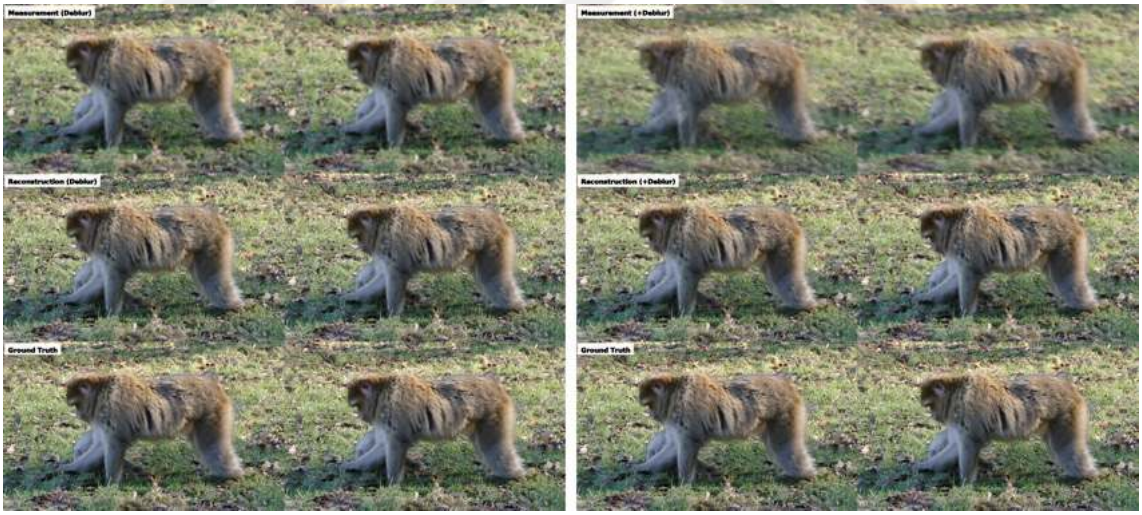

Figure 4.12: Representative comparison of two sequential frames from **DAVIS** dataset, **Left: Deblur, right: Deblur+** tasks. From top to bottom: measurement, reconstructed by **Baseline** (VISION-XL), and ground truth.

Table 4.1: Quantitative evaluation on **DAVIS** dataset comparing **Baseline** and **Baseline+** across various degradation tasks. Best values are **bolded**.

Degradation	Method	PSNR↑	SSIM↑	MSE↓	NIQE↓	DISTS↓	ERQA↑	LPIPS Alex↓	LPIPS VGG↓	fvd-videogpt↓	fvd-styleganv↓	flow_MSE↓	Straightness↑	P-ST↓	D-ST↓
sr	Baseline	29.6809	0.8271	145.7962	**6.0304**	0.1428	0.4827	**0.2223**	0.2415	84.0918	83.4553	**0.0140**	**95.490**	**0.1334**	159.8273
	Baseline+	**29.7367**	**0.8294**	**144.9656**	6.1289	**0.1414**	**0.4839**	0.2249	**0.2381**	**83.8501**	**82.2478**	0.0145	95.454	0.1350	**159.4358**
+sr	Baseline	28.5554	0.7871	163.2963	**6.2104**	0.1683	0.4565	**0.2861**	0.3419	**117.7656**	**118.1167**	**0.0148**	**92.376**	**0.1774**	**178.1258**
	Baseline+	**28.5687**	**0.7878**	**163.0102**	6.2499	**0.1676**	**0.4569**	0.2869	**0.3418**	122.4390	120.0357	0.0151	92.322	0.1781	178.1407
deblur	Baseline	31.1156	0.8525	118.2038	6.5971	0.1301	0.5684	0.2304	0.2649	73.4979	71.6275	**0.0122**	**95.652**	0.1380	130.4136
	Baseline+	**31.1342**	**0.8529**	**117.9922**	**6.5927**	**0.1298**	**0.5697**	**0.2302**	**0.2644**	**69.1178**	**68.6272**	0.0123	95.616	**0.1379**	**130.3061**
+deblur	Baseline	**28.1384**	**0.7736**	**173.1263**	6.6623	0.1831	0.4020	**0.3470**	**0.3855**	171.4347	**166.8834**	**0.0156**	**92.538**	**0.2149**	**188.7698**
	Baseline+	28.1199	0.7735	173.6303	**6.6476**	**0.1826**	**0.4035**	0.3473	0.3856	**169.0877**	167.1298	0.0159	92.250	0.2157	189.5531
inpainting	Baseline	29.1401	0.8005	163.6318	5.1775	0.1884	0.6687	0.2251	0.3513	120.0330	114.1284	0.0143	**94.788**	0.1360	177.9612
	Baseline+	**29.2978**	**0.8071**	**160.6640**	**5.1247**	**0.1791**	**0.6727**	**0.2128**	**0.3432**	**101.5679**	**100.9374**	**0.0137**	94.428	**0.1291**	**174.3881**
+inpainting	Baseline	28.1125	0.7729	180.8796	5.4023	0.2067	0.6429	0.2700	0.3894	171.9401	169.9593	**0.0146**	**92.844**	0.1666	195.5216
	Baseline+	**28.1422**	**0.7750**	**179.9522**	**5.3892**	**0.2043**	**0.6441**	**0.2663**	**0.3876**	**166.3593**	**165.0876**	0.0148	92.754	**0.1645**	**194.7402**
temporal blur	Baseline	30.4425	0.7938	146.4884	**4.5700**	**0.1152**	0.7386	0.2151	0.2930	**50.5858**	50.1056	0.0128	**94.194**	0.1308	159.2915
	Baseline+	**30.4446**	**0.7938**	**146.4743**	4.5720	0.1153	**0.7387**	**0.2149**	**0.2930**	51.4089	**50.0101**	**0.0127**	94.158	0.1308	**159.1931**

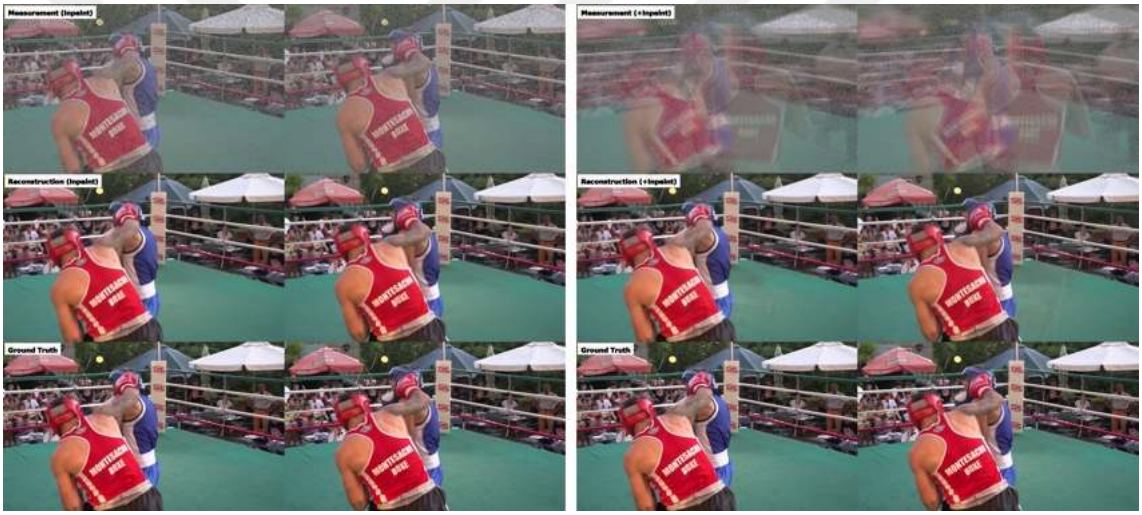

Figure 4.13: Representative comparison of two sequential frames from **DAVIS** dataset, **Left: Inpaint, right: Inpaint+** tasks. From top to bottom: measurement, reconstructed by **Baseline** (VISION-XL), and ground truth.

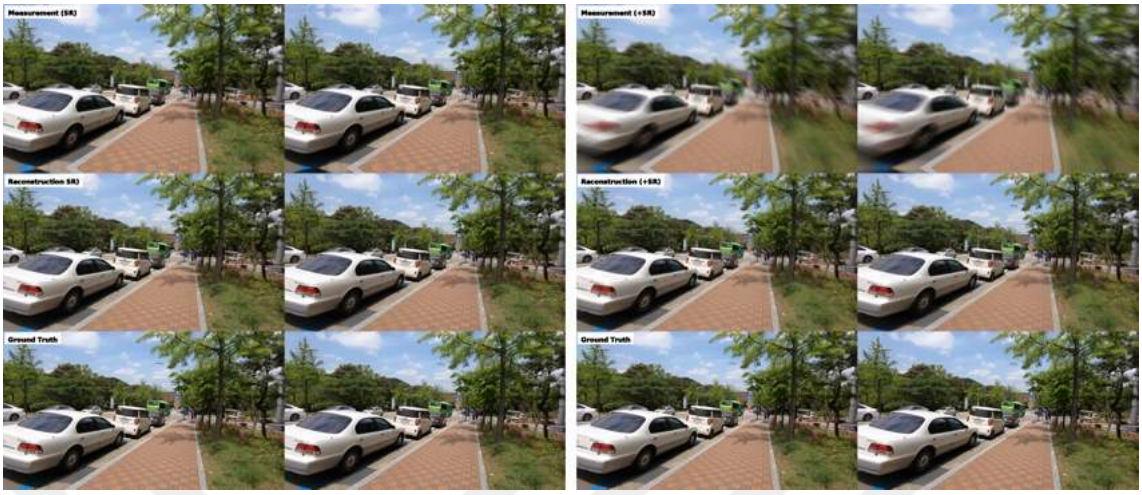

Figure 4.14: Representative comparison of two sequential frames from **REDS4** dataset, **Left: SR, right: SR+** tasks. From top to bottom: measurement, reconstructed by **Baseline** (VISION-XL), and ground truth.

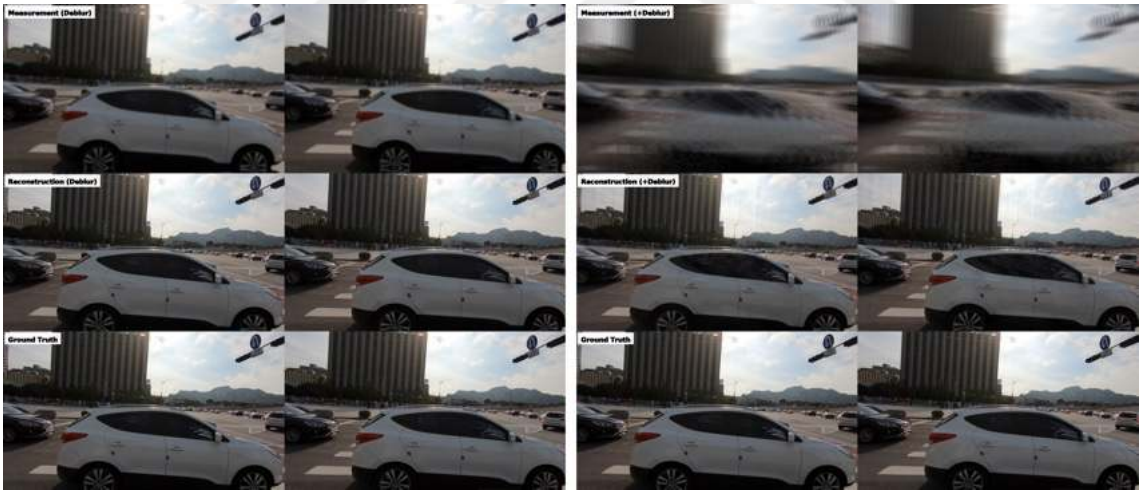

Figure 4.15: Representative comparison of two sequential frames from **REDS4** dataset, **Left: Deblur, right: Deblur+** tasks. From top to bottom: measurement, reconstructed by **Baseline** (VISION-XL), and ground truth.

Figure 4.16: Representative comparison of two sequential frames from **REDS4** dataset, **Left: Inpaint, right: Inpaint+** tasks. From top to bottom: measurement, reconstructed by **Baseline** (VISION-XL), and ground truth.

Table 4.2: Quantitative evaluation on **REDS4** dataset comparing **Baseline** and **Baseline+** across various degradation tasks. Best values are **bolded**.

Degradation	Method	PSNR↑	SSIM↑	MSE↓	NIQE↓	DISTS↓	ERQA↑	LPIPS Alex↓	LPIPS VGG↓	fvd-videogpt↓	fvd-styleganv↓	flow_MSE↓	Straightness↑	P-ST↓	D-ST↓
sr	Baseline	26.2504	0.7413	165.2900	**5.2643**	0.1881	**0.4516**	**0.3130**	**0.3101**	53.6515	53.9013	0.0135	**103.644**	**0.1730**	178.8216
	Baseline+	**26.2763**	**0.7431**	**164.2261**	5.3510	**0.1878**	0.4499	0.3195	0.3108	**51.6672**	**51.8888**	**0.0135**	103.626	0.1767	**177.7404**
+sr	Baseline	25.1724	0.6873	211.5076	**5.8083**	0.2207	0.3879	**0.3738**	0.3928	158.4851	156.6652	**0.0149**	**98.244**	**0.2180**	226.4110
	Baseline+	**25.2008**	**0.6887**	**210.0749**	5.8415	**0.2194**	**0.3884**	0.3758	**0.3922**	**141.7857**	**140.7089**	0.0149	98.082	0.2195	**225.0139**
deblur	Baseline	27.6862	0.7906	121.0771	6.6310	0.1622	0.5392	0.2997	0.2823	50.0600	49.9828	0.0118	**103.986**	0.1652	132.9215
	Baseline+	**27.7020**	**0.7913**	**120.6692**	**6.6048**	**0.1615**	**0.5406**	**0.2996**	**0.2817**	**46.7897**	**46.8866**	**0.0117**	103.932	**0.1651**	**132.4128**
+deblur	Baseline	25.1437	0.6730	214.3348	6.2234	0.2262	0.3426	0.4247	0.4308	161.3587	161.6394	0.0158	**98.856**	0.2461	230.1515
	Baseline+	**25.1720**	**0.6741**	**212.8786**	**6.2111**	**0.2247**	**0.3457**	**0.4235**	**0.4300**	**147.5224**	**147.4916**	**0.0153**	98.658	**0.2460**	**228.1795**
inpainting	Baseline	26.4958	0.7395	163.0164	4.1169	0.1906	0.6672	0.1915	0.3198	98.5989	98.6108	**0.0129**	**102.276**	0.1073	175.9154
	Baseline+	**26.6221**	**0.7468**	**158.3607**	**4.0148**	**0.1811**	**0.6703**	**0.1843**	**0.3140**	**97.6533**	**95.8653**	0.0131	101.862	**0.1037**	**171.4974**
+inpainting	Baseline	24.9886	0.6738	225.5475	4.5324	0.2447	0.5906	0.2797	0.3996	313.5906	312.3551	0.0147	**97.668**	0.1641	240.2639
	Baseline+	**25.0411**	**0.6775**	**222.7862**	**4.5086**	**0.2407**	**0.5920**	**0.2752**	**0.3977**	**299.5098**	**298.4670**	**0.0146**	97.524	**0.1617**	**237.4179**
temporal blur	Baseline	26.6409	0.7321	159.8196	3.6175	0.1044	0.6886	0.1638	0.2931	**206.8385**	**206.7784**	0.0136	**98.514**	0.0953	173.4345
	Baseline+	**26.6432**	**0.7323**	**159.8017**	**3.6108**	**0.1042**	**0.6890**	**0.1635**	**0.2928**	207.6850	207.3517	**0.0133**	98.424	**0.0952**	**173.1191**

Based on the quantitative results reported in Table 4.1 for DAVIS dataset and Table 4.2 for REDS4 dataset across almost all degradations (except deblur+), **Baseline+**, which employs frame-specific latent initialization, shows marginal improvements in fidelity metrics. For example, for DAVIS dataset, in the `sr` and `inpaint`, PSNR increased from 29.68 to 29.74 and 29.14 to 29.30, respectively, and SSIM and MSE also improved slightly, showing marginal enhancement in pixel-level accuracy. In perceptual metrics, **Baseline+** achieves better results for DISTS and LPIPS-VGG across almost all restoration tasks. Other perceptual metrics like NIQE,

LPIPS-Alex, and ERQA show mixed results. FlowMSE and perceptual straightness do not improve, while FVD metrics fluctuate across different tasks.

Overall, the most noticeable benefit of **Baseline+** is across spatial-only degradations, while temporal degradations (`+sr`, `+deblur`, `+inpainting`) show mixed results, indicating that **Baseline+** maintains or even improves the results if the measurement frames themselves are not severely degraded by motion blur.

4.6.3 Ablation Study: Effect of Cross-Frame Attention

Objective. We want to integrate cross-frame attention (CFA) into the SDXL UNet of VISION-XL and evaluate whether it improves temporal coherence and perceptual quality while preserving fidelity. We compare three CFA strategies discussed in Section 4.3.5 versus the Baseline+ which uses self-attention with no temporal coupling.

Experimental setup We integrate CFA into SDXL by replacing the original attention processors in self-attention blocks with a `CrossFrameAttnProcessor` according to Algorithm 1 which computes queries from the current frame and uses keys/values from a reference bank. We examine two different policies:

Experiment1: CFA-First (first-frame anchor) All frames attend to the key/value features of the first frame (Equation 4.32). This method gives a global temporal anchor and has a constant memory cost.

Experiment2: CFA-Window (windowed, exponentially-weighted) Frame n attends to a window of the last w frames including current frame (**??**) with exponentially decaying weights for blending the keys/values in order to balance drift and over-smoothing.

To isolate the effect of CFA, we keep other things like the sampling and optimization schedule identical to Baseline+ (we used frame-specific DDIM inversion for latent initialization) All SDXL weights are frozen and we only swap the attention processors for each experiment.

Experimental Results and Analysis We reported the evaluation results of Experiment 1 (CFA-First) for DAVIS and REDS4 test datasets in Table 4.3 and Table 4.4 respectively. As it is shown in the tables, using the K/V of the first frame as a global anchor mostly improves video-level quality for some tasks. Across DAVIS and REDS4, Straightness increases in nearly all degradations, meaning that motion looks smoother and more predictable. FVD also improves for pure temporal blur (clear drops on both datasets), SR (small but consistent gains), and inpainting (strong on REDS4) and for +inpainting (on both datasets). In contrast, for deblur and +deblur FVD can be slightly worse, even though Straightness still rises. Per-frame fidelity metrics (PSNR/SSIM, DISTS, LPIPS) tend to worsen slightly with CFA-First, showing a small cost in sharpness/texture matching.

Task-wise interpretation

Table 4.3: Quantitative evaluation on **DAVIS** dataset comparing **Cross-Frame Attention** and **Baseline** across various degradation tasks. Best values are **bolded**.

Degradation	Method	PSNR↑	SSIM↑	MSE↓	NIQE↓	DISTS↓	ERQA↑	LPIPS Alex↓	LPIPS VGG↓	fvd-videogpt↓	fvd-styleganv↓	flow_MSE↓	Straightness↑	P-ST↓	D-ST↓
sr	CFA	29.6987	0.8285	145.3834	**6.1278**	0.1429	0.4821	0.2256	0.2393	**83.0042**	**82.1097**	**0.0141**	**95.490**	0.1354	**159.4353**
	Baseline+	**29.7367**	**0.8294**	**144.9656**	6.1289	**0.1414**	**0.4839**	**0.2249**	**0.2381**	83.8501	82.2478	0.0145	95.454	**0.1350**	159.4358
+sr	CFA	28.5149	0.7855	163.8802	6.2599	0.1703	0.4546	0.2899	0.3452	124.2781	123.7754	0.0151	**92.808**	0.1790	179.0212
	Baseline+	**28.5687**	**0.7878**	**163.0102**	**6.2499**	**0.1676**	**0.4569**	**0.2869**	**0.3418**	**122.4390**	**120.0357**	**0.0151**	92.322	**0.1781**	**178.1407**
deblur	CFA	**31.1352**	**0.8529**	**117.9777**	6.6230	0.1300	0.5694	**0.2297**	**0.2633**	69.9283	68.9115	**0.0122**	**95.652**	**0.1376**	**130.1879**
	Baseline+	31.1342	0.8529	117.9922	**6.5927**	**0.1298**	**0.5697**	0.2302	0.2644	**69.1178**	**68.6272**	0.0123	95.616	0.1379	130.3061
+deblur	CFA	28.1171	0.7722	173.9217	6.7089	0.1840	0.4020	0.3487	0.3863	172.5570	169.5180	0.0162	**93.060**	**0.2147**	190.1102
	Baseline+	**28.1199**	**0.7735**	**173.6303**	**6.6476**	**0.1826**	**0.4035**	**0.3473**	**0.3856**	**169.0877**	**167.1298**	**0.0159**	92.250	0.2157	**189.5531**
inpainting	CFA	29.1691	0.8038	161.9879	5.1487	0.1849	0.6704	0.2208	0.3504	102.8234	**100.6821**	0.0140	**94.626**	0.1337	176.0373
	Baseline+	**29.2978**	**0.8071**	**160.6640**	**5.1247**	**0.1791**	**0.6727**	**0.2128**	**0.3432**	**101.5679**	100.9374	**0.0137**	94.428	**0.1291**	**174.3881**
+inpainting	CFA	28.0454	0.7702	181.3859	5.4289	0.2110	0.6408	0.2754	0.3952	**164.1281**	**163.9457**	**0.0147**	**93.834**	0.1682	196.0635
	Baseline+	**28.1422**	**0.7750**	**179.9522**	**5.3892**	**0.2043**	**0.6441**	**0.2663**	**0.3876**	166.3593	165.0876	0.0148	92.754	**0.1645**	**194.7402**
temporal*blur	CFA	**30.4535**	**0.7939**	**146.3659**	**4.5593**	**0.1149**	**0.7393**	**0.2142**	**0.2922**	**49.0240**	**48.1801**	0.0129	**94.248**	**0.1302**	159.2769
	Baseline+	30.4446	0.7938	146.4743	4.5720	0.1153	0.7387	0.2149	0.2930	51.4089	50.0101	**0.0127**	94.194	0.1308	**159.1931**

Table 4.4: Quantitative evaluation on **REDS4** dataset comparing **Cross-Frame Attention** and **Baseline** across various degradation tasks. Best values are **bolded**.

Degradation	Method	PSNR↑	SSIM↑	MSE↓	NIQE↓	DISTS↓	ERQA↑	LPIPS Alex↓	LPIPS VGG↓	fvd-videogpt↓	fvd-styleganv↓	flow_MSE↓	Straightness↑	P-ST↓	D-ST↓
sr	CFA	26.1924	0.7388	167.1834	5.3652	0.1911	0.4449	0.3226	0.3171	**49.7157**	**48.5144**	**0.0133**	**103.680**	0.1783	180.4679
	Baseline+	**26.2763**	**0.7431**	**164.2261**	**5.3510**	**0.1878**	**0.4499**	**0.3195**	**0.3108**	51.6672	51.8888	0.0135	103.626	**0.1767**	**177.7404**
+sr	CFA	25.0761	0.6821	215.7129	5.9102	0.2250	0.3822	0.3833	0.4022	157.7117	158.5108	**0.0146**	**98.424**	0.2231	230.3555
	Baseline+	**25.2008**	**0.6887**	**210.0749**	**5.8415**	**0.2194**	**0.3884**	**0.3758**	**0.3922**	**141.7857**	**140.7089**	0.0149	98.082	**0.2195**	**225.0139**
deblur	CFA	27.6921	0.7905	121.0108	6.7211	0.1619	0.5397	0.3003	0.2830	48.9468	48.0800	0.0118	**103.986**	0.1655	132.8442
	Baseline+	**27.7020**	**0.7913**	**120.6692**	**6.6048**	**0.1615**	**0.5406**	**0.2996**	**0.2817**	**46.7897**	**46.8866**	**0.0117**	103.932	**0.1651**	**132.4128**
+deblur	CFA	25.0797	0.6697	217.2162	6.3161	0.2278	0.3397	0.4286	0.4341	156.5287	156.0115	0.0155	**98.946**	0.2482	232.7046
	Baseline+	**25.1720**	**0.6741**	**212.8786**	**6.2111**	**0.2247**	**0.3457**	**0.4235**	**0.4300**	**147.5224**	**147.4916**	**0.0153**	98.658	**0.2460**	**228.1795**
inpainting	CFA	26.5260	0.7418	160.9774	4.1512	0.1893	0.6676	0.1932	0.3247	**86.3712**	**86.6619**	0.0133	**101.97**	0.1085	174.2734
	Baseline+	**26.6221**	**0.7468**	**158.3607**	**4.0148**	**0.1811**	**0.6703**	**0.1843**	**0.3140**	97.6533	95.8653	**0.0131**	101.862	**0.1037**	**171.4974**
+inpainting	CFA	24.8894	0.6689	229.4985	4.6440	0.2500	0.5870	0.2870	0.4098	**296.6266**	**295.1489**	0.0151	**98.37**	0.1672	244.6195
	Baseline+	**25.0411**	**0.6775**	**222.7862**	**4.5086**	**0.2407**	**0.5920**	**0.2752**	**0.3977**	299.5098	298.4670	**0.0146**	97.524	**0.1617**	**237.4179**
temporal*blur	CFA	26.5522	0.7286	162.6710	**3.5856**	0.1066	0.6874	0.1653	0.2967	**194.2880**	**193.3250**	0.0139	**99.054**	0.0956	176.5510
	Baseline+	**26.6432**	**0.7323**	**159.8017**	3.6108	**0.1042**	**0.6890**	**0.1635**	**0.2928**	207.6850	207.3517	**0.0133**	98.424	**0.0952**	**173.1191**

Interpretation: Our temporal blur uses circular averaging; frame 1 is not globally clean. However, as illustrated in Figure 4.17, for degradations with temporal blur, in low-motion, stable regions, and non-overlapping regions the averaging does less damage to fine structures around the first frame than around interior frames (neighboring wrap-around frames are only slightly displaced there). As diffusion progresses, the first-frame latent at each timestep becomes cleaner, so its K/V tokens preserve these less-ruined structures. Reusing them across the sequence reduces drift and flicker in those stable areas, which is exactly what **FVD** and **Straightness** reward. This explains the consistent FVD drops for blur. (For SR, less micro-flicker in invented details, and for (+)inpainting filled regions keep stable color/pattern over time).

When the task also needs strong spatial restoration (e.g. deblurring or upsampling while also handling temporal blur), a single, fixed anchor can stabilize the wrong reference: it anchors the sequence to the coarse features of the first frame and suppresses valid frame-specific updates such as new edges, dis-occlusions, and rapid texture changes. The result is smoother trajectories (↑Straightness) but distributional mismatch in deep video features (↑FVD).

limitation: The effect of CFA-First depends on how informative the first frame is. If large parts of frame 1 are heavily blurred or occluded, those regions provide weak K/V tokens and can propagate biased textures in later frames. Conversely, when first frame contains relatively sharper structures, the anchor is helpful and we observe better FVD and Straightness.

Figure 4.17: left: the reference first measurement frame, right: an arbitrary measurement frame in that sequence, selected from **REDS4** dataset and **temporal blur** degradation

For completeness, we also tested a *windowed CFA* (concatenating several recent frames with decay). While, it sometimes reduced LPIPS/NIQE, it **did not improve temporal coherence** in our setup and occasionally **hurt FVD/Straightness**, likely because unaligned past tokens introduce mild motion lag on fast scenes.

4.6.4 Ablation Study: Effect of Perceptual Straightening

To evaluate the effectiveness of different integration methods of perceptual straightening, we did comparative experiments using three optimization strategies as discussed in Section 4.4.3.

Experimental Setup We implemented three optimization strategies:

Experiment 1: Sequential Optimization: The sequential approach implements a two-stage optimization pipeline where data consistency refinement precedes perceptual straightening optimization according to Algorithm 3

According to VISION-XL pipeline, in each sampling step, after decoding the estimated clean latent sequence to pixel-space, we first apply $N_{\text{DDS}} = 5$ conjugate gradient iterations for data consistency (DDS), then to enforce perceptual-straightness we compute perceptual straightening loss and apply $N_{\text{PS}} = 10$ gradient descent steps with learning rate $\lambda_{\text{PS}} = 0.5$. This approach has potential risk of deteriorating data consistency after DDS step.

Experiment 2: Alternating Optimization:

According to Algorithm 4, this approach alternates between single conjugate gradient steps for data consistency and single gradient descent steps for perceptual straightening for $N_{\text{alt}} = 10$ iterations using the same learning rate ($\eta_{PS} = 0.5$). The potential advantage of this strategy is avoiding significant drift from both objectives by continuously correcting deviations.

Experiment 3: Joint Optimization: Following Algorithm 5, in this experiment we simultaneously optimize both objectives using Adam optimizer with normalized loss scaling. Data fidelity and perceptual straightening losses are normalized by their initial values to ensure balanced optimization ($N_{joint} = 20$ iterations, $lr = 0.01$, $\lambda_{PS} = 1.0$).

Following the experiment setup we used for baseline, all experiments were done on the same set of video sequences with both spatial and spatio-temporal degradations.

Experimental Results and Analysis We evaluate three integration strategies for perceptual straightening (PS) within VISION-XL and report detailed quantitative results for the sequential variant (Experiment 1) in Table 4.5 and 4.6. Across DAVIS and REDS4 and for all degradations, the *Straightness* score improves comparing to *Baseline+*, evidence that using Perceptual Straightening guidance straightens the V1-trajectory for the restored videos. Consistently, PSNR/SSIM, LPIPS, and Flow-MSE is not improved but stays on par with Baseline+, indicating that perceptual straightening optimization is primarily a temporal regularizer.

More interestingly, our analysis reveals a clear dependency between task complexity and perceptual straightening effectiveness. When degradations contain tem-

poral blur (**sr+**, **deblur+**, **temporal-blur**), the improvement in Straightness metric is more pronounced and Straightness metric correlates with FVD across both datasets. One explanation for this effect is that Straightness specifically measures the smoothness of frame-to-frame evolution in a perceptual space, and our perceptual straightening guidance primarily improves temporal coherence. In motion-blur-like degradations the dominant error is temporal (suppressed or smeared motion cues); correcting it both straightens the perceptual trajectory and reduces FVD. In contrast, for spatial-only degradations, temporal dynamics are largely preserved; gains in Straightness reflect minor flicker reduction but do not address the spatial realism that mainly determines FVD, so the two metrics are not correlated.

Table 4.5: Quantitative evaluation on **DAVIS** dataset comparing **Perceptual Straightening** optimization versus **Baseline+** across various degradation tasks. Best values are **bolded**.

Degradation	Method	PSNR↑	SSIM↑	MSE↓	NIQE↓	DISTS↓	ERQA↑	LPIPS Alex↓	LPIPS VGG↓	fvd-videogpt↓	fvd-styleganv↓	flow_MSE↓	Straightness↑	P-ST↓	D-ST↓
sr	PS	29.6799	0.8271	145.8019	**6.0284**	0.1428	0.4826	**0.2223**	0.2415	85.9838	83.3062	**0.0142**	**95.760**	**0.1330**	159.9752
	Baseline+	**29.7367**	**0.8294**	**144.9656**	6.1289	**0.1414**	**0.4839**	0.2249	**0.2381**	**83.8501**	**82.2478**	0.0145	95.454	0.1350	**159.4358**
+sr	PS	28.5560	0.7871	163.2954	**6.2095**	0.1683	0.4563	**0.2861**	0.3419	**117.1282**	**116.1784**	0.0151	**92.790**	**0.1766**	178.4411
	Baseline+	**28.5687**	**0.7878**	**163.0102**	6.2499	**0.1676**	**0.4569**	0.2869	**0.3418**	122.4390	120.0357	**0.0151**	92.322	0.1781	**178.1407**
deblur	PS	31.1147	0.8525	118.2078	6.6008	0.1300	0.5682	0.2303	0.2649	70.8423	70.7495	0.0126	**95.940**	**0.1376**	130.8105
	Baseline+	**31.1342**	**0.8529**	**117.9922**	**6.5927**	**0.1298**	**0.5697**	**0.2302**	**0.2644**	**69.1178**	**68.6272**	**0.0123**	95.616	0.1379	**130.3061**
+deblur	PS	**28.1387**	**0.7736**	**173.1204**	6.6618	0.1831	0.4019	**0.3471**	**0.3855**	**164.5605**	**164.5999**	0.0159	**93.096**	**0.2136**	**189.0676**
	Baseline+	28.1199	0.7735	173.6303	**6.6476**	**0.1826**	**0.4035**	0.3473	0.3856	169.0877	167.1298	**0.0159**	92.25	0.2157	189.5531
inpainting	PS	29.1369	0.8004	163.6802	5.1794	0.1884	0.6686	0.2251	0.3514	112.9809	110.3423	0.0138	**95.256**	0.1354	177.4314
	Baseline+	**29.2978**	**0.8071**	**160.6640**	**5.1247**	**0.1791**	**0.6727**	**0.2128**	**0.3432**	**101.5679**	**100.9374**	**0.0137**	94.428	**0.1291**	**174.3881**
+inpainting	PS	28.1106	0.7729	180.9264	5.4011	0.2067	0.6431	0.2700	0.3895	172.1538	168.4197	**0.0142**	**93.618**	0.1653	195.1133
	Baseline+	**28.1422**	**0.7750**	**179.9522**	**5.3892**	**0.2043**	**0.6441**	**0.2663**	**0.3876**	**166.3593**	**165.0876**	0.0148	92.754	**0.1645**	**194.7402**
temporal blur	PS	30.4432	0.7938	146.4857	**4.5684**	**0.1152**	0.7386	0.2151	0.2931	**49.5639**	**48.8714**	**0.0127**	**94.410**	**0.1305**	**159.1802**
	Baseline+	**30.4446**	**0.7938**	**146.4743**	4.5720	0.1153	**0.7387**	**0.2149**	**0.2930**	51.4089	50.0101	0.0127	94.194	0.1308	159.1931

Table 4.6: Quantitative evaluation on **REDS4** dataset comparing **Perceptual Straightening** optimization versus **Baseline+** across various degradation tasks. Best values are **bolded**.

Degradation	Method	PSNR↑	SSIM↑	MSE↓	NIQE↓	DISTS↓	ERQA↑	LPIPS Alex↓	LPIPS VGG↓	fvd-videogpt↓	fvd-styleganv↓	flow_MSE↓	Straightness↑	P-ST↓	D-ST↓
sr	PS	26.2507	0.7413	165.2860	**5.2627**	0.1882	**0.4518**	**0.3130**	**0.3101**	54.0367	52.8803	**0.0131**	**108.644**	**0.1730**	178.3603
	Baseline+	**26.2763**	**0.7431**	**164.2261**	5.3510	**0.1878**	0.4499	0.3195	0.3108	**51.6672**	**51.8888**	0.0135	103.626	0.1767	**177.7404**
+sr	PS	25.1730	0.6873	211.4596	**5.8042**	0.2206	0.3879	**0.3737**	0.3928	**140.0593**	**139.8951**	**0.0147**	**98.298**	**0.2178**	226.1452
	Baseline+	**25.2008**	**0.6887**	**210.0749**	5.8415	**0.2194**	**0.3884**	0.3758	**0.3922**	141.7857	140.7089	0.0149	98.082	0.2195	**225.0139**
deblur	PS	27.6859	0.7906	121.0905	6.6324	0.1623	0.5392	0.2998	0.2824	50.6460	50.3979	0.0121	**103.968**	0.1652	133.2403
	Baseline+	**27.7020**	**0.7913**	**120.6692**	**6.6048**	**0.1615**	**0.5406**	**0.2996**	**0.2817**	**46.7897**	**46.8866**	**0.0117**	103.932	**0.1651**	**132.4128**
+deblur	PS	25.1441	0.6730	214.3020	6.2271	0.2260	0.3426	0.4246	0.4307	**147.1632**	**147.1276**	0.0157	**98.856**	0.2461	230.0379
	Baseline+	**25.1720**	**0.6741**	**212.8786**	**6.2111**	**0.2247**	**0.3457**	**0.4235**	**0.4300**	147.5224	147.4916	**0.0153**	98.658	**0.2460**	**228.1795**
inpainting	PS	-	-	-	-	-	-	-	-	-	-	-	-	-	-
	Baseline+	**26.6221**	**0.7468**	**158.3607**	**4.0148**	**0.1811**	**0.6703**	**0.1843**	**0.3140**	**97.6533**	**95.8653**	**0.0131**	**101.862**	**0.1037**	**171.4974**
+inpainting	PS	25.0361	0.6773	222.9363	**4.4996**	0.2409	**0.5921**	0.2759	0.3980	320.8514	320.3053	0.0152	**97.524**	0.1621	238.1001
	Baseline+	**25.0411**	**0.6775**	**222.7862**	4.5086	**0.2407**	0.5920	**0.2752**	**0.3977**	**299.5098**	**298.4670**	**0.0146**	**97.524**	**0.1617**	**237.4179**
temporal*blur	PS	26.6410	0.7321	159.8291	3.6202	0.1046	0.6888	0.1638	0.2932	**195.7243**	**194.9824**	0.0135	**98.514**	0.0953	173.3719
	Baseline+	**26.6432**	**0.7323**	**159.8017**	**3.6108**	**0.1042**	**0.6890**	**0.1635**	**0.2928**	207.6850	207.3517	**0.0133**	98.424	**0.0952**	**173.1191**

Since perceptual straightening acts as a temporal regularizer, we expect it to leave frame appearance largely unchanged while making motion look more natural by small edits. As Figures 4.18 and 4.19 depicts, using perceptual straightening guidance, we observe reduced micro-wobble in structures like building edges, window frames, less boiling of repetitive textures and cleaner line persistence.

Figure 4.18: comparison of **Perceptual Straightening Optimization** versus Baseline+ on **REDS** dataset

Figure 4.19: comparison of **Perceptual Straightening Optimization** versus Baseline+ on **REDS** dataset

We also explored alternative enforcement schemes for perceptual straightening explained in Experiment 2 and experiment 3, but neither leads to satisfactory results.

In Experiment 2 (alternating optimization), although the curvature is reduced resulting in improvement in Straightness metric, but this resulted in temporally blured results and fidelity loss specially for degradations contain temporal blur. In this strategy, the PS gradient is computed and applied when the frames are still far from the natural-image manifold because of the degradation. In that state the retinal–V1 features are unstable, so minimizing curvature tends to reduce the frame-to-frame variation rather than preserve motion-consistent variation. This behaves like a temporal low-pass filter: it straightens the trajectory in perceptual space and (undesirably) straightens in pixel space, and this effect is amplified under temporally blurred degradations where features are already misaligned. Moreover, single-step alternating between DDS and PS optimization creates gradient interference: DDS enforces toward measurement-consistent details while PS—computed on off-manifold frames—penalizes the temporal changes needed to reconstruct sharp, dynamic content, so PS step undoes what DDS just restored. Because curvature

loss reduces quickly by "making neighbors look alike" and results in smooth but less faithful videos (good straightness number, worse FVD/PSNR). In Experiment 3 (joint optimization), beside the same problem occured in Experiment 2, we observed strong sensitivity to hyper parameters causing instability or collapse in most of the cases. Overall, applying PS optimization before frames are sufficiently "clean" and aligned may make "straighter" trajectories but not perceptually natural motion; this explains why alternating/joint optimization mechanisms increased degraded visual quality, whereas the sequential schedule kept PS within its validity region.

4.6.5 Ablation Study: Effect of Multi-Path Ensemble Sampling

To evaluate the effectiveness of our proposed multi-path ensemble sampling framework discussed in Section 4.5 , we did comprehensive ablation studies to examine different ensemble strategies discussed in 4.5.3.

Experiment 1: Synchronized Multi-Path Sampling Latent-Space and Pixel-space Fusion (K=2) We implement a synchronized sampling strategy explianed in Algorithm 7, where two trajectories share information during the denoising process. We want to see whether continuous latent-space synchronization combined with final pixel-space fusion can achieve better results compared to the Baseline. Both paths start from identical latents initialized using frame-specific DDIM inversion similar to Baseline+: $z_\tau^{(1)} = z_\tau^{(2)} = z_\tau$. After processing both trajectories at timestep t using the Vision-XL pipeline, we fuse their noisy latent representations using uniform averaging: $z_{fused} = \frac{1}{2}(z_t^{(1)} + z_t^{(2)})$. This fused latent is then used as the starting point for both trajectories at the next timestep: $z_{t-1}^{(1)} = z_{t-1}^{(2)} = z_{fused}$. This synchronization continues throughout the sampling process. At the final timestep, we decode the clean latent from the synchronized trajectories to the pixel space separately and then compute uniform averaging of the reconstructed images: $X_{ensemble} = 0.5 \times X_0^{(1)} + 0.5 \times X_0^{(2)}$. To ensure data consistency with the measurement, we apply a final conjugate gradient (CG) optimization step on final fused frames. This strategy enforces both trajectories to remain close to each other while still to be able to explore slightly different regions of the solution

space.

Experiment 2: Independent Multi-Path Sampling, Latent-Space Fusion (K=2). We run two completely independent diffusion samplings, each performing the standard Vision-XL sampling initialized similar to Experiment 1. The trajectories are independent throughout the all denoising timesteps and do not share any information. At the final timestep, we fuse the denoised latents before VAE decoding: $\hat{z}_{ensemble} = \frac{1}{2}(\hat{z}_0^{(1)} + \hat{z}_0^{(2)})$, then decode the fused latent to pixel space: $X_{ensemble} = D_\theta(\hat{z}_{ensemble})$. To avoid deteriorating data consistency with the measurement, we apply a final conjugate gradient (CG) optimization step on final fused frames.

Experiment 3: Independent Multi-Path Sampling, Pixel-Space Fusion (K=2). Similar to Experiment 2, we generate two independent trajectories through the entire sampling process, but this time we do the fusion strategy in the pixel space. In fact, at the final timestep, we decode both latents into the pixel space using the VAE decoder and fuse them through uniform averaging: $X_{ensemble} = \frac{1}{2}(X_0^{(1)} + X_0^{(2)})$. This experiment allows direct comparison with Experiment 2 to find the optimal fusion representation space (latent vs. pixel) for ensemble sampling.

Experiment 4: Independent Multi-Path Sampling, Pixel-Space Fusion (K=3). We extend Experiment 3 to three independent trajectories, keeping the same uniform averaging strategy only at the final step and in the pixel domain: $X_{ensemble} = \frac{1}{3}(X_0^{(1)} + X_0^{(2)} + X_0^{(3)})$. This experiment assesses whether ensembling more paths provides additional benefits over the two-path ensemble because of the theoretical analysis that show averaging more independent estimators should reduce variance further according to the law of large numbers.

These experiments evaluate three key aspects of ensemble sampling, we discussed in methodology: (1) the effect of ensemble size (K=2 vs K=3), (2) the ideal fusion representation space (latent vs pixel), and (3) the timing of information sharing (end-only vs every-step synchronization). These experiments provide insights into the most effective ensemble strategies for zero-shot diffusion-based video restoration.

Figure 4.20: Representative comparison of **Ensemble sampling** experiments for **SR** task on **DAVIS** dataset

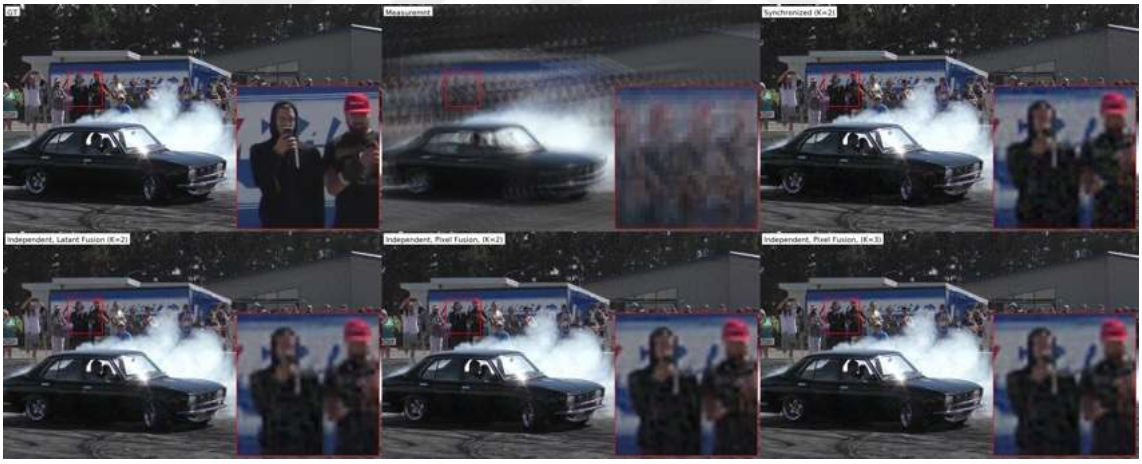

Figure 4.21: Representative comparison of **Ensemble sampling** experiments for **+SR** task on **DAVIS** dataset

Figure 4.22: Representative comparison of **Ensemble sampling** experiments for **Deblur** task on **DAVIS** dataset

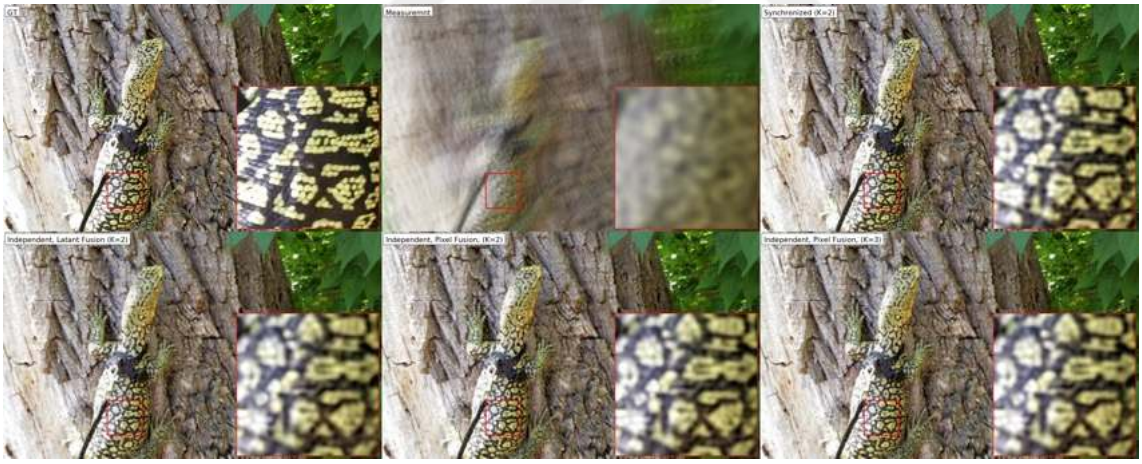

Figure 4.23: Representative comparison of **Ensemble sampling** experiments for **+Deblur** task on **DAVIS** dataset

Figure 4.24: Representative comparison of **Ensemble sampling** experiments for **+Inpainting** task on **DAVIS** dataset

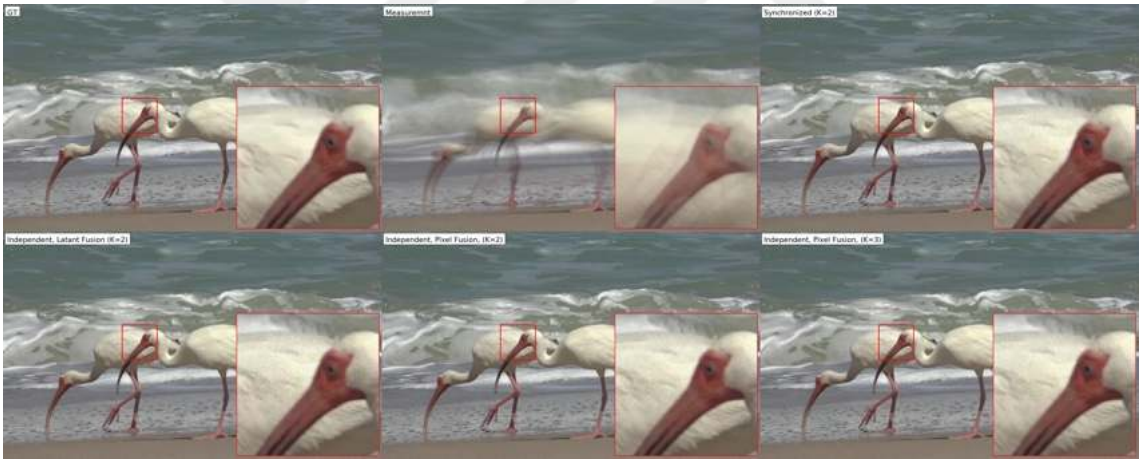

Figure 4.25: Representative comparison of **Ensemble sampling** experiments for Temporal Blur task on **DAVIS** dataset

Experimental Results and Analysis To evaluate ensemble sampling strategies, we performed same four experiments on two different datasets. Tables 4.7 and 4.8 illustrate the performance comparison for DAVIS and REDS4 datasets respectively, displaying consistent patterns and also dataset-specific characteristics that we will go over them.

- **Synchronized vs Independent Ensemble Sampling** Independent ap-

Table 4.7: Quantitative evaluation on **DAVIS** dataset comparing different **Ensemble** ablation experiments across various degradation tasks. Best values are **bolded** and blue values represent the second-best scores

Degradation	Method	PSNR↑	SSIM↑	MSE↓	NIQE↓	DISTS↓	ERQA↑	LPIPS Alex↓	LPIPS VGG↓	fvd-videogpt↓	fvd-styleganv↓	flow_MSE↓	Straightness↑	P-ST↓	D-ST↓
sr	Synchronized (K=2)	30.4239	0.8540	131.9721	**5.5561**	**0.1255**	**0.5429**	**0.1945**	0.2143	**67.1925**	**66.6535**	0.0136	96.066	**0.1160**	145.5508
	Independent, Latant Fusion (K=2)	30.2939	0.8520	132.9672	5.6751	0.1304	0.5304	0.1963	0.2214	70.5963	70.3234	0.0137	96.048	0.1171	146.6228
	Independent, Pixel Fusion, (K=2)	30.4857	0.8558	130.9022	5.7019	0.1259	0.5391	0.1988	0.2119	67.5485	67.6531	0.0134	96.084	0.1185	144.3505
	Independent, Pixel Fusion, (K=3)	**30.5181**	**0.8566**	**130.4066**	5.7677	0.1262	0.5378	0.2011	**0.2111**	69.4153	68.6170	**0.0134**	**0.5338**	0.1199	**143.7916**
	Baseline+	29.7367	0.8294	144.9656	6.1289	0.1414	0.4839	0.2249	0.2381	83.8501	82.2478	0.0145	95.454	0.1350	159.4358
+sr	Synchronized (K=2)	28.7856	0.7598	153.0000	**5.3697**	0.1725	**0.5410**	0.3278	0.3749	92.1276	92.5761	0.0145	91.602	0.2050	167.4869
	Independent, Latant Fusion (K=2)	28.9560	0.8003	150.7103	5.4961	0.1640	0.5138	**0.2633**	0.3419	99.8820	98.6967	0.0148	92.250	0.1635	165.5139
	Independent, Pixel Fusion, (K=2)	29.2106	0.8076	146.5153	5.6564	0.1563	0.5220	0.2659	0.3278	92.6848	91.0386	0.0144	92.844	0.1641	160.9432
	Independent, Pixel Fusion, (K=3)	**29.2684**	**0.8114**	**145.5845**	5.7581	**0.1552**	0.5201	0.2643	**0.3225**	**91.3272**	**90.1016**	**0.0142**	**92.952**	**0.1629**	**159.7503**
	Baseline+	28.5687	0.7878	163.0102	6.2499	0.1676	0.4569	0.2869	0.3418	122.4390	120.0357	0.0151	92.322	0.1781	178.1407
deblur	Synchronized (K=2)	25.7401	0.5499	214.1702	7.9810	0.2292	0.4205	0.5031	0.4600	155.1677	154.5224	0.0122	94.086	0.3064	226.3626
	Independent, Latant Fusion (K=2)	32.4332	0.8780	96.6836	**6.1444**	0.1313	0.6270	0.2196	0.2508	64.6716	64.7970	**0.0108**	96.390	0.1305	107.5168
	Independent, Pixel Fusion, (K=2)	32.5858	0.8807	95.0014	6.1676	0.1287	**0.6391**	0.2196	0.2388	**63.7341**	**64.3902**	0.0111	96.390	0.1305	106.0960
	Independent, Pixel Fusion, (K=3)	**32.6534**	**0.8834**	**94.6855**	6.1907	**0.1280**	0.6389	**0.2163**	**0.2335**	64.4063	64.5867	0.0111	**96.392**	**0.1286**	**105.7445**
	Baseline+	31.1342	0.8529	117.9922	6.5927	0.1298	0.5697	0.2302	0.2644	69.1178	68.6272	0.0123	95.616	0.1379	130.3061
+deblur	Synchronized (K=2)	26.3283	0.5710	203.5108	7.8060	0.2243	0.4354	0.4849	0.4759	169.3116	166.0150	**0.0142**	92.214	0.3013	217.6775
	Independent, Latant Fusion (K=2)	28.5169	0.7770	160.0142	**6.3212**	0.1766	0.4493	0.3312	0.3851	147.8147	146.2795	0.0150	92.340	0.2055	174.9943
	Independent, Pixel Fusion, (K=2)	28.7287	0.7768	154.2380	6.3384	0.1732	**0.4742**	0.3358	0.3732	142.6238	139.6994	0.0146	93.150	0.2065	168.7994
	Independent, Pixel Fusion, (K=3)	**28.7903**	**0.7818**	**153.3616**	6.3522	**0.1721**	0.4734	**0.3309**	**0.3682**	**142.5426**	**139.6423**	0.0149	**93.222**	**0.2034**	**168.2588**
	Baseline+	28.1199	0.7735	173.6303	6.6476	0.1826	0.4035	0.3473	0.3856	169.0877	167.1298	0.0159	92.250	0.2157	189.5531
inpainting	Synchronized (K=2)	-	-	-	-	-	-	-	-	-	-	-	-	-	-
	Independent, Latant Fusion (K=2)	31.1709	0.8682	91.7836	6.8056	0.1902	0.7558	0.1900	0.3239	46.2753	46.2289	0.0128	94.536	0.1151	104.5946
	Independent, Pixel Fusion, (K=2)	32.2791	0.8927	81.8521	6.2681	0.1613	0.7774	0.1483	0.2896	37.3911	**36.4159**	0.0124	94.950	0.0895	94.2314
	Independent, Pixel Fusion, (K=3)	**32.4059**	**0.8962**	**79.9219**	6.1423	**0.1573**	**0.7775**	**0.1408**	**0.2823**	**35.4686**	36.9999	**0.0124**	**0.5278**	**0.0849**	**92.2860**
	Baseline+	29.2978	0.8071	160.6640	**5.1247**	0.1791	0.6727	0.2128	0.3432	101.5679	100.9374	0.0137	94.428	0.1291	174.3881
+inpainting	Synchronized (K=2)	29.8057	0.7873	110.5180	7.3924	0.2172	**0.7574**	0.2655	0.3651	**73.2573**	**73.5369**	**0.0130**	93.456	0.1628	123.5403
	Independent, Latant Fusion (K=2)	29.5276	0.8191	116.7221	6.9567	0.2169	0.7260	0.2446	0.3707	102.8738	102.8408	0.0133	92.610	0.1513	130.0616
	Independent, Pixel Fusion, (K=2)	30.3436	0.8461	105.0209	6.4748	0.1905	0.7487	0.2048	0.3423	91.1018	89.7861	0.0132	93.402	0.1256	118.2630
	Independent, Pixel Fusion, (K=3)	**30.4528**	**0.8508**	**102.8256**	6.3212	**0.1863**	0.7486	**0.1968**	**0.3368**	89.6017	89.1968	0.0133	**93.510**	**0.1205**	**116.1095**
	Baseline+	28.1422	0.7750	179.9522	**5.3892**	0.2043	0.6441	0.2663	0.3876	166.3593	165.0876	0.0148	92.754	0.1645	194.7402
temporal blur	Synchronized (K=2)	33.9308	0.8202	30.3164	5.5142	0.1239	0.8607	0.1729	0.2386	16.9505	16.2265	0.0110	94.014	0.1054	41.3614
	Independent, Latant Fusion (K=2)	33.5976	0.8168	32.8499	5.4338	0.1261	0.8552	0.1772	0.2440	17.2757	16.9627	0.0115	93.564	0.1085	44.3696
	Independent, Pixel Fusion, (K=2)	34.7168	0.8469	25.4745	5.2299	0.1118	0.8762	0.1481	0.2220	12.0802	12.5348	**0.0109**	93.978	0.0903	36.4243
	Independent, Pixel Fusion, (K=3)	**35.1709**	**0.8583**	**22.9810**	5.1405	**0.1062**	**0.8834**	**0.1369**	**0.2135**	**10.4322**	**10.6899**	0.0111	94.068	**0.0834**	**34.0602**
	Baseline+	30.4446	0.7938	146.4743	**4.5720**	0.1153	0.7387	0.2149	0.2930	51.4089	50.0101	0.0127	**94.194**	0.1308	159.1931

Table 4.8: Quantitative evaluation on **REDS4** dataset comparing different **Ensemble** ablation experiments across various degradation tasks. Best values are **bolded**.

Degradation	Method	PSNR↑	SSIM↑	MSE↓	NIQE↓	DISTS↓	ERQA↑	LPIPS Alex↓	LPIPS VGG↓	fvd-videogpt↓	fvd-styleganv↓	flow_MSE↓	Straightness↑	P-ST↓	D-ST↓
sr	Synchronized (K=2)	26.7289	0.7728	147.5101	**4.7594**	**0.1686**	**0.5183**	**0.2774**	0.2840	50.4917	**49.9254**	0.0125	104.436	**0.1522**	159.9946
	Independent, Latant Fusion (K=2)	26.6506	0.7697	149.9132	5.0066	0.1728	0.5060	0.2774	0.2928	55.8290	54.7760	0.0126	104.400	0.1523	162.5172
	Independent, Pixel Fusion, (K=2)	26.7709	0.7747	146.0941	4.9310	0.1702	0.5141	0.2839	0.2834	50.7844	51.1504	0.0124	104.454	0.1557	158.5174
	Independent, Pixel Fusion, (K=3)	**26.7912**	**0.7756**	**145.4140**	5.0098	0.1709	0.5124	0.2870	**0.2831**	**50.1927**	50.8866	**0.0124**	**104.472**	0.1574	**157.7888**
	Baseline+	26.2763	0.7431	164.2261	5.3510	0.1878	0.4499	0.3195	0.3108	51.6672	51.8888	0.0135	103.626	0.1767	177.7404
+sr	Synchronized (K=2)	**25.7746**	0.7047	**182.4461**	**5.1324**	**0.1981**	**0.4985**	0.3397	0.3886	**102.3603**	**102.1864**	**0.0141**	98.766	0.1971	**196.5544**
	Independent, Latant Fusion (K=2)	25.5295	0.7063	193.9907	5.3644	0.2064	0.4550	**0.3366**	0.3860	135.6013	135.3260	0.0143	98.424	**0.1960**	208.2861
	Independent, Pixel Fusion, (K=2)	25.7114	0.7158	186.1943	5.3732	0.1997	0.4635	0.3421	0.3716	123.7890	123.7817	0.0145	98.910	0.1982	200.6943
	Independent, Pixel Fusion, (K=3)	25.7314	**0.7175**	185.3616	5.4540	0.2002	0.4614	0.3450	**0.3696**	124.5278	124.0970	0.0143	**98.964**	0.1997	199.7032
	Baseline+	25.2008	0.6887	210.0749	5.8415	0.2194	0.3884	0.3758	0.3922	141.7857	140.7089	0.0149	98.082	0.2195	225.0139
deblur	Synchronized (K=2)	26.0150	0.6251	166.3833	7.1181	0.1810	0.5905	0.3633	0.3792	88.1390	88.0435	**0.0110**	103.806	0.2005	177.4160
	Independent, Latant Fusion (K=2)	28.7609	0.8308	93.4338	5.9514	0.1547	0.6002	0.2912	0.2561	54.5014	52.9646	0.0112	105.029	0.1588	104.6369
	Independent, Pixel Fusion, (K=2)	28.9183	0.8361	90.3736	**5.9123**	**0.1526**	**0.6127**	**0.2880**	0.2463	51.8873	53.1965	0.0111	105.029	**0.1571**	101.4920
	Independent, Pixel Fusion, (K=3)	**28.9350**	**0.8378**	**90.0950**	5.9346	0.1534	0.6121	0.2893	**0.2442**	52.5554	52.6310	0.0111	**105.030**	0.1578	**101.1511**
	Baseline+	27.7020	0.7913	120.6692	6.6048	0.1615	0.5406	0.2996	0.2817	**46.7897**	**46.8866**	0.0117	103.932	0.1651	132.4128
+deblur	Synchronized (K=2)	24.4821	0.5315	234.9195	7.4135	0.2277	**0.5099**	0.4478	0.4832	153.3266	153.3061	0.0143	99.180	0.2591	249.2054
	Independent, Latant Fusion (K=2)	25.5242	0.6810	195.6159	6.0540	0.2125	0.3973	0.4088	0.4249	134.2759	133.3326	**0.0142**	98.928	0.2368	209.8171
	Independent, Pixel Fusion, (K=2)	25.7344	0.6873	186.0343	**6.0460**	**0.2066**	0.4248	**0.4039**	0.4108	126.7702	125.8548	0.0146	99.414	**0.2328**	200.6415
	Independent, Pixel Fusion, (K=3)	**25.7601**	**0.6910**	**185.1467**	6.0617	0.2067	0.4235	0.4045	**0.4078**	**126.4900**	**125.6219**	0.0148	**99.486**	0.2330	**199.9967**
	Baseline+	25.1720	0.6741	212.8786	6.2111	0.2247	0.3457	0.4235	0.4300	147.5224	147.4916	0.0153	98.658	0.2460	228.1795
inpainting	Synchronized (K=2)	-	-	-	-	-	-	-	-	-	-	-	-	-	-
	Independent, Latant Fusion (K=2)	28.8064	0.8481	93.9859	5.3304	0.1868	0.7727	0.1495	0.2783	38.7122	38.4632	0.0121	102.240	0.0838	106.0495
	Independent, Pixel Fusion, (K=2)	29.5643	0.8694	80.3047	4.9554	0.1622	**0.7867**	0.1208	0.2475	**32.4185**	**32.8247**	0.0120	102.816	0.0673	92.2713
	Independent, Pixel Fusion, (K=3)	**29.6793**	**0.8731**	**78.2358**	4.7748	**0.1586**	0.7863	**0.1166**	**0.2421**	34.0742	33.1742	**0.0119**	**102.870**	**0.0649**	**90.1356**
	Baseline+	26.6221	0.7468	158.3607	**4.0148**	0.1811	0.6703	0.1843	0.3140	97.6533	95.8653	0.0131	101.862	0.1037	171.4974
+inpainting	Synchronized (K=2)	26.9395	0.7726	143.6372	5.7186	0.2307	0.7245	0.2211	0.3547	191.9295	190.1561	0.0137	98.226	0.1290	157.3515
	Independent, Latant Fusion (K=2)	26.3208	0.7509	164.9390	5.6832	0.2469	0.6934	0.2478	0.3812	229.5450	228.7902	0.0144	97.632	0.1454	179.3182
	Independent, Pixel Fusion, (K=2)	**27.0310**	0.7733	**140.4356**	6.1478	0.2371	**0.7430**	0.2217	0.3534	**176.1535**	**176.1139**	**0.0134**	98.280	0.1292	**153.8649**
	Independent, Pixel Fusion, (K=3)	26.9266	**0.7801**	144.6385	5.0780	**0.2199**	0.7087	**0.2112**	**0.3506**	206.0452	205.7182	0.0138	**98.352**	**0.1230**	158.4105
	Baseline+	25.0411	0.6775	222.7862	**4.5086**	0.2407	0.5920	0.2752	0.3977	299.5098	298.4670	0.0146	97.524	0.1617	237.4179
temporal blur	Synchronized (K=2)	**30.9716**	0.8511	**54.5824**	2.9513	0.1034	**0.8523**	0.1088	**0.2192**	**124.1220**	**123.4700**	**0.0127**	99.270	0.0628	**67.2384**
	Independent, Latant Fusion (K=2)	30.1218	0.8552	69.2092	**2.9064**	0.1036	0.8201	0.1215	0.2385	143.9532	143.6348	0.0136	98.640	0.0706	82.7805
	Independent, Pixel Fusion, (K=2)	30.5346	0.8660	63.4579	2.9575	0.0929	0.8273	0.1110	0.2250	138.6359	138.4109	0.0127	99.414	0.0639	76.1517
	Independent, Pixel Fusion, (K=3)	30.6237	**0.8699**	62.2899	2.9538	**0.0907**	0.8282	**0.1085**	0.2217	143.9935	142.2372	0.0128	**99.504**	**0.0625**	75.0959
	Baseline+	26.6432	0.7323	159.8017	3.6108	0.1042	0.6890	0.1635	0.2928	207.6850	207.3517	0.0133	98.424	0.0952	173.1191

proaches consistently outperform synchronized multi-path sampling (Experiment 1) for both datasets. Most importantly, synchronized sampling shows unstable characteristics. For example, it fails in inpainting task for both datasets possibly because the fused latents are outside the range that the VAE decoder expects. On other tasks, synchronized sampling shows considerable deterioration on challenging inverse problems: achieving only 25.74 dB PSNR on DAVIS deblurring vs. 32.65 dB for independent pixel fusion.

This poor performance possibly occurred because the synchronization of latents eliminates the trajectory diversity and both paths converge to suboptimal solutions. However, as showed in the visual comparison results in Figures 4.20 to 4.25 synchronized ensembling sometimes achieves better perceptual metrics, specially for SR/SR+ tasks (5.56 vs 5.77 NIQE on SR for DAVIS) by restoring high frequency details.

-Latent vs Pixel Space Fusion Independent pixel-space fusion methods (Experiment3, Experiment4) generally excel latent-space fusion (Experiment 1, Experiment2) over different tasks and datasets. On DAVIS inpainting task, for example, Experiment 3 achieves 32.28 dB PSNR versus 31.17 dB in Experiment 2. This superiority potentially occurs because latent-space fusion loses high-frequency information during the single decoding step of averaged latent representations.

-Ensemble Size Effects Increasing independent trajectories from K=2 to K=3 slightly improves the results across both datasets, specifically for spatial fidelity metrics. DAVIS shows gains of 0.4-0.5 dB PSNR (temporal blur: 34.72→35.17 dB) and REDS4 exhibits similar improvements (30.53→30.62 dB). As discussed in section4.5.4, these improvements, specifically in distortion metrics, align with ensemble theory that states averaging more independent estimators reduces variance. However, the computational cost increases linearly, making K=2 more proper in practice.

-Cross-Dataset evaluation Performance is consistent for DAVIS and REDS4, but REDS4 shows 3-4 dB lower PSNR values because of the complexity of motion patterns in it. In general, independent pixel fusion (K=3) (Experiment 4) shows better metrics on both datasets, while synchronized sampling (Experiment 1) demonstrates instability and deteriorated performance.

-Perception-Distortion Trade-offs Looking to the evaluation results, synchronized sampling shows poor performance on challenging tasks (deblurring), across both perceptual and distortion metrics. On moderate tasks (e.g. super-resolution) synchronized sampling may achieve competitive perceptual scores (NIQE, DISTS) through local texture generation, but based on the visual results illustrated in Figures 4.20 to 4.25 global spatial and temporal inconsistencies manifest as artifacts. To find the method with best perception-distortion balance, Figures 4.26 and 4.27 depict the perception-distortion trade-off based on P_{ST} (spatio-temporal perceptual quality) and D_{ST} (spatio-temporal distortion) metrics which are proposed in Chapter 3.

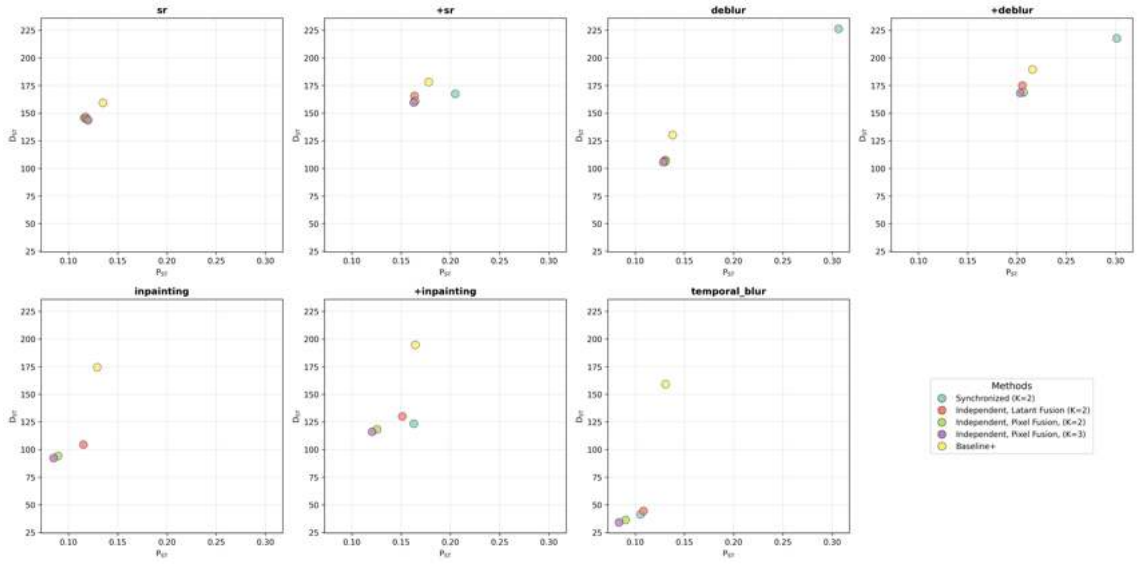

Figure 4.26: Perception-Distortion trade-off for different Ensemble methods and Baseline+ on **DAVIS**

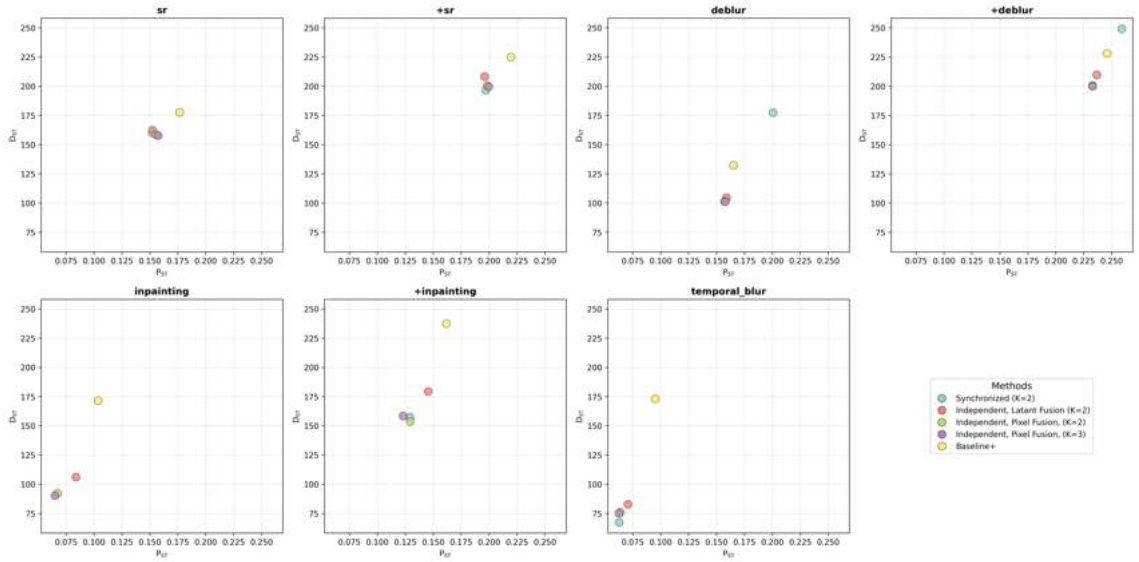

Figure 4.27: Perception-Distortion trade-off for different Ensemble methods and Baseline+ on **REDS4**

Baseline vs Ensemble Method Comparison We compare all four ensemble strategies against the baseline Vision-XL initialized with frame-based DDIM inversion (Baseline+) since the same initialization strategy is used in all Ensemble experiments. Table 4.7 and Table 4.8 present the comparison results for DAVIS and REDS4 datasets.

-Universal Ensemble Improvements: All ensemble methods outperform Baseline+ across almost all restoration tasks and metrics. For example, on DAVIS deblurring, although the worst-performing ensemble method (Synchronized K=2: 25.74 dB) fails dramatically, all independent methods considerably outperform Baseline+: Independent Latent K=2 (32.43 dB), Independent Pixel K=2 (32.59 dB), and Independent Pixel K=3 (32.65 dB) all surpass Baseline+ (31.13 dB) by 1.3-1.5 dB. This consistent improvement indicates that trajectory diversity substantially improves reconstruction quality.

-Task-Dependent Improvement Magnitudes: By investigating the results, we can see that the benefit of ensemble sampling is correlated with task difficulty. Simple tasks like SR show moderate 0.5-0.8 dB improvement for PSNR across all ensemble methods, while challenging tasks show huge gains: deblurring improve-

ments range from 1.3-1.5 dB, and temporal degradations achieve 3.2-4.7 dB gains. This observation shows the importance of ensembling where inverse problem solvers face limitations with single path.

Figure 4.28 shows the comparison results of Independent Pixel Fusion (k=2) versus Baseline+ for all restoration tasks. As it is shown in the results, ensemble method improved the restoration quality for all tasks.

Figure 4.28: Visual comparison of **Experiment 3** (Independent pixel fusion (k=2)) versus **Baseline+** on **DAVIS** for different degradations

-Ensemble Method Generalizability In spite of different content characteristics of DAVIS and REDS4 and different degradation types, ensemble methods show consistent improvement over Baseline+. This observation strongly validates the generalization ability of our ensemble sampling framework.

-Computational overhead Independent pixel-fusion with K=2 provides the best balance across both datasets. It achieves on par performance with Independent pixel-fusion with K=3, while requiring only around double (rather than triple) of the computational overhead of Baseline+.

The results indicate that independent pixel fusion ensemble sampling has significant and consistent improvements over single-path Vision-XL baseline. Specifically, for challenging inverse problems where single path sampling may lead to suboptimal solutions, the role of trajectory diversity becomes more crucial. The K=2 variant offers the best practical trade-off between performance gains and computational efficiency.

4.7 Summary and Conclusion

The proposed ensembling strategy shows that averaging multiple diffusion trajectories leads to consistent gains in both spatial fidelity, perceptual quality, and temporal coherence across diverse restoration tasks. Ensembling approaches reduce stochastic artifacts and stabilize frame evolution without requiring additional training or model modifications. Experimental results confirm improvements in perceptual metrics and robustness, highlighting ensembling as a simple, yet effective inference-time strategy. Nonetheless, the added computational overhead suggests a trade-off between efficiency and quality, which could be further optimized in future work.

Chapter 5

DISCUSSION AND CONCLUSIONS

5.1 Summary of Contributions and Key Findings

This thesis investigates video restoration under two complementary paradigms (task-specific supervised models (Chapter 3) and zero-shot large generative priors (Chapter 4)). The contributions of this thesis include: (1) a new GAN-based PVSR model with two discriminators, where a flow discriminator encourages naturalness of motion; (2) Using Perceptual Straightening Hypothesis (PSH) to design a metric for motion naturalness and spatio-temporal perceptual metric; (3) Using perceptual straightening optimization as guidance within zero-shot diffusion-based VSR to improve motion naturalness; (4) investigating Cross-Frame attention mechanisms within zero-shot diffusion-based VSR; (5) multi-path ensembling that aggregates multiple stochastic diffusion trajectories and suppress frame-to-frame texture jitter and improves the overall quality.

5.2 Attention — Discussion & Future Work

We integrated cross-frame attention (CFA) into the SDXL UNet of VISION-XL in a training-free manner. Our main variant "first-frame CFA" fixes the key/value bank to the first frame, providing a global temporal anchor across the sequence. Empirically, this method improves video-level coherence (higher Straightness, lower FVD in several tasks), at the cost of slightly worse single-frame fidelity (PSNR/SSIM, LPIPS/DISTS).

Limitations.

- **Anchor dependence.** first-frame anchored CFA inherits the strengths and weaknesses of the first frame. If the anchor frame contains heavy degradation (blur/occlusion), CFA can propagate harmful or biased textures.

- **Texture locking:** A fixed anchor stabilizes motion, but it resists lawful changes (new edges, dis-occlusions, rapid texture variation), which is most visible in Deblur / SR+ / Deblur+.

- **Per-frame fidelity trade-off.** Slight degradations in spatial distortion and perceptual scores indicate a small cost in sharpness/texture matching when enforcing temporal stability.

Future work.

We suggest some future works that can be explored to further benefit from cross-frame attention mechanism.

- **Anchor selection and refresh:** Replacing "first-frame only" with sharpest-frame, refreshing it periodically every k frames, or using mixture of anchors (first + median) to reduce the bias can be explored by future work.

- **Motion-/occlusion-aware gating (training-free).** Down-weighting CFA where flow magnitude or temporal variance is high or where occlusion masks show new content and using the anchor in low-motion regions may improve the performance.

- **Integrating to different Layers:** To integrate cross-frame attention only in down/mid/up blocks of UNet and let other blocks to remain self-attentive to allow diversity and local detail formation.

- **alignment before attention:** Future work can explore warping $(\mathbf{K}, \mathbf{V})$ to the current frame using coarse flow (or deformable offsets) to reduce key competition from misaligned tokens.

5.3 Perceptual Straightening — Discussion & Future Work

This thesis exploits the Perceptual Straightening Hypothesis (PSH) in two complementary ways. (1) In Chapter 3, we formalized a temporal naturalness metric by mapping videos into a retinal–V1 representation and measuring trajectory "Straightness". We showed that Straightness algins better with perceived motion naturalness in VSR evaluations. (2) In Chapter 4, we used Straightness as an optimization guidance inside VISION-XL: in each denoising step, we refine predicted clean frames with a curvature penalty computed in the retinal–V1 space. Evaluation results showed that this method consistently improved Straightness for all datasets and degradations and improves FVD specifically when degradations contained temporal blur (e.g., `sr+`, `deblur+`, temporal-blur).

Despite the benefits, Our investigations reveal some limitations for Straightness metric and PS optimization method.

Limitations of the Straightness metric

- **Representation dependence:** Straightness is computed in the RetinaND/V1 embedding space. degradations like camera shake can perturb these features independently of human judgment.

- **Non-causality:** Computing Straightness requires aggregating the representations over sequences; it's not causal and this limits the use for real-time feedback or streaming evaluation.

Limitations of the Straightness optimization

- **Helps mostly when the error is temporal:** Straightening improves "temporal naturalness" and often correlates with FVD only when degradations include temporal blur. For pure spatial degradations, Straightness improves while other metrics do not, so benefits to overall quality can be limited.

- **Compute overhead:** PS optimization requires extra gradient steps in each

sampling step; Although we do PS refinement in the pixel-space optimization phase (After DDS) to avoid repeated latent decodes, but the added iterations still increase runtime (around 12 seconds latency for 25-frame video on Tesla-V100).

- **Feature instability early in sampling:** In early denoising steps or strong degradations when frames are still far from the natural-image manifold, V1 features are unstable and minimizing curvature can cause over-smoothing (the problem observed in Alternating and joint PS optimization)

Future Directions

Following are some possible directions for future work:

- **Better Perceptual Embedders:** Future work can adopt feature spaces that were shown to separate natural and AI-generated video trajectories such as those used in perceptual-straightening based detectors as the backbone [Internò et al., 2025] for both the metric and the guidance.

- **Partial-path guidance:** Future work can apply perceptual-straightening guidance only in the mid/late denoising steps or on shorter frame sequences to avoid over-smoothing.

- **Better curvature penalty term:** One can adopt margin-based (hinge/Huber) curvature penalties with adaptive thresholds and a scheduler that increases PS update weight as samples approach the image manifold.

5.4 Multi-Path Ensemble Sampling — Discussion & Future Work

Our experiments in Section 4.6.5 demonstrate that ensemble strategies consistently improve spatio-temporal perceptual quality and distortion metrics for a variety of video restoration tasks comparing to the baseline. Using the diversity of independent

sampling paths, ensemble methods help reduce stochastic artifacts and stabilize frame restoration without requiring retraining.

Why Averaging Works in Practice

In principle, averaging multiple posterior samples should approximate the MMSE estimator, which often is blurry. However, our ensemble results are not blurred and both distortion and perceptual metrics improve simultaneously. The reason would be several factors:

1. All trajectories begin from the same noisy latent and only diverge due to stochasticity during denoising steps. This makes them highly correlated, so averaging reinforces consistent structures rather than collapsing into a blurry MMSE estimate.

2. In video restoration tasks, the posterior distribution is concentrated around the natural data manifold. When only a few trajectories are averaged, they remain close to each other, so the mean preserves major structures instead of washing them out.

3. Each trajectory may contain small hallucinations or random details. Averaging across paths suppresses these inconsistencies, leading to cleaner and more stable reconstructions.

Despite the benefits, several limitations and open challenges remain related to multi-path ensemble sampling.

Limitations and Challenges

The main limitation of ensemble methods is their computational overhead. Running multiple restorations increases inference time or GPU memory usage, which may limit real-time or resource-constrained applications. Moreover, our results show diminishing gains beyond a certain number of ensemble members, suggesting a trade-off between quality gains and efficiency. Another challenge is the lack of principled

fusion strategies. Our current averaging scheme is simple but may not combine individual paths optimally.

Future Directions

There are several promising avenues to extend this work:

- **Advanced Fusion Strategies:** Developing adaptive fusion mechanisms, such as attention-based or quality-aware weighting mechanisms that dynamically combine results based on reconstruction confidence.
- **Generalization Studies:** Testing multi-path ensemble sampling using different diffusion architectures (e.g. Stable Diffusion 1.5 [Rombach et al., 2022b], pix-art [Chen et al., 2023], Consistency Models) and broader domains.
- **Partial-Path Ensembling:** Instead of running K full trajectories, we could run K trajectories only for some initial denoising steps (e.g., 30–50% of the denoising process). At this point, we could fuse the trajectories and the single fused state can be refined by continuing denoising. This approach reduces computational cost while still using diversity in early steps to stabilize reconstruction.

5.5 Task-Specific GAN versus Zero-Shot LDMs for VSR

In this section, we briefly compare task-specific supervised models of Chapter 3 with zero-shot large-scale generative models of Chapter 4, and we conclude by outlining the main limitations of the current work and concise directions for future research. On the task-specific side, we adopt a perceptual VSR (PVSR) architecture which has a generator and two discriminators: a spatial (texture) discriminator and a temporal (motion) discriminator, we initialize the generator with a VSR model trained on REDS dataset (EDVR and BasicVSR++) and finetune the generator on this dataset

using a combination of losses; this yields sharper frames and improves motion naturalness for super resolution task, but for restoration tasks such as deblurring or inpainting, requires per-task retraining and finetuning. In contrast, our zero-shot approach based on a diffusion-based inverse problem solver (VISION-XL), uses a general text-to-image generative model (SDXL) as the prior and it can be used for general purpose video restoration and in Chapter 4 we showed that we can improve its performance in a training-free manner and only using inference time strategies.

Using the evaluation results for super resolution task in Table 5.1, we compare a zero-shot model developed in Chapter 4 (Independent, Pixel Fusion, (K=2)) with the task-specific BasicVSR++ from Chapter 3. On **REDS4** test dataset degraded with bicubic downsampling (consistent with degradation used for training dataset), **BasicVSR++** dominates the zero-shot method (e.g., PSNR 32.39 vs. 26.77), indicating the advantage of supervised models under matched distribution. However, when the degradation kernel changes to average pooling during inference, the task-specific model (trained with bicubic kernel) degrades markedly on REDS4 (PSNR 23.91), whereas the **zero-shot** method remains stronger. A similar pattern happens under **dataset shift** restoring **DAVIS** dataset, where the zero-shot approach shows better performance. In short, *BasicVSR++* excels in-domain with matched degradations and similar data distribution, whereas the *Independent, Pixel Fusion (K=2)* zero-shot ensemble offers improved robustness to data/degradation shifts.

Table 5.1: Quantitative evaluation on comparing our zero-shot **Ensemble** method and task-specific **BasicVSR++** for **SR** task on **DAVIS** and **REDS4** datasets. Best values are **bolded**.

Dataset	Method	PSNR↑	SSIM↑	MSE↓	NIQE↓	DISTS↓	ERQA↑	LPIPS Alex↓	LPIPS VGG↓	fvd-videogpt↓	fvd-styleganv↓	flow_MSE↓	Straightness↑	P-ST↓	D-ST↓
Davis (SR)	Independent, Pixel Fusion, (K=2) (AvgPool)	**30.49**	0.8558	**130.90**	5.7019	**0.1259**	0.5391	0.1988	**0.2119**	67.5485	67.6531	0.0134	**96.0805**	0.1185	**144.3505**
	BasicVSR++(AvgPool)	29.64	**0.8714**	159.26	**5.5405**	0.1281	**0.6669**	**0.1880**	0.2137	**59.8630**	**61.2268**	**0.0128**	95.7029	**0.1125**	172.1016
REDS4 (SR)	Independent, Pixel Fusion, (K=2) (AvgPool)	26.77	0.7747	146.09	4.9310	0.1702	0.5141	0.2839	0.2834	50.7844	51.1504	0.0124	104.4613	0.1557	158.5174
	BasicVSR++(Bicubic)	**32.39**	**0.907**	**43.89**	**3.9090**	**0.1098**	**0.7890**	**0.1302**	**0.1726**	**16.9721**	**17.3581**	**0.0102**	**104.5204**	**0.0714**	**54.0470**
	BasicVSR++(AvgPool)	23.91	0.7526	272.93	4.7061	0.1706	0.6339	0.2822	0.3022	81.6456	81.2002	0.0130	103.6715	0.1560	285.8894

As shown in Figure 5.1, when we apply BasicVSR++ to inputs degraded with AveragePooling rather than the bicubic downsampling it was trained on, a domain mismatch occurs and the model hallucinates sharp details and edges more aggressively, producing outputs that appear extra sharp but contain noticeable artifacts

and textures that do not match the ground truth.

Figure 5.1: Visual comparison of **Independent, Pixel Fusion, (K=2)** and task-specific **BasicVSR++** on **REDS4** for SR task

Diffusion-based zero-shot restoration has substantial latency and compute overhead due to iterative denoising and data-consistency steps, which is amplified by ensembling. On the same hardware (Tesla V100), our baseline *VISION-XL* takes ∼2 minutes to process a 25-frame clip (ensemble methods with K=2 take ∼ 2× longer, i.e. about ∼4 minutes), compared with ∼45 seconds for *BasicVSR++*.

5.6 Conclusion.

This thesis provided a thorough investigation of perceptual video restoration using two complementary paradigms: task-specific supervised models and zero-shot diffusion-based generative priors. We demonstrated how temporal coherence and perceptual quality may be enhanced under various restoration scenarios by adding novel architectures, cross-frame attention methods,perceptual straightening guidance, multi-path ensembling. Our evaluations also showed that while task-specific models are superior when tested under training-matched conditions, zero-shot diffu-

sion approaches offer a flexible, training-free alternative with promising adaptability. Together, these findings highlight both the advantages and the limitations of current work, and suggest directions to future research for scalable, temporally coherent video restoration.

BIBLIOGRAPHY

[Bao et al., 2023] Bao, Y., Qiu, D., Kang, G., Zhang, B., Jin, B., Wang, K., and Yan, P. (2023). Latentwarp: Consistent diffusion latents for zero-shot video-to-video translation. *arXiv preprint arXiv:2311.00353*.

[Bishop and Nasrabadi, 2006] Bishop, C. M. and Nasrabadi, N. M. (2006). *Pattern recognition and machine learning*, volume 4. Springer.

[Blau and Michaeli, 2018] Blau, Y. and Michaeli, T. (2018). The perception-distortion tradeoff. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6228–6237.

[Caballero et al., 2017] Caballero, J., Ledig, C., Aitken, A. ., Acosta, A., Totz, J., Wang, Z., and Shi, W. (2017). Realtime video super resolution with spatio temporal networks and motion compensation. In *IEEE/CVF Conf. on Computer Vision and Pattern Recognition*.

[Cao et al., 2025] Cao, C., Yue, H., Liu, X., and Yang, J. (2025). Zero-shot video restoration and enhancement using pre-trained image diffusion model. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 1935–1943.

[Chan et al., 2021] Chan, K. C., Wang, X., Yu, K., Dong, C., and Loy, C. C. (2021). Basicvsr: The search for essential components in video super-resolution and beyond. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4947–4956.

[Chan et al., 2022] Chan, K. C., Zhou, S., Xu, X., and Loy, C. C. (2022). Basicvsr++: Improving video super-resolution with enhanced propagation and align-

ment. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5972–5981.

[Chang et al., 2004] Chang, H., Yeung, D.-Y., and Xiong, Y. (2004). Super-resolution through neighbor embedding. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, volume 1, pages I–I. IEEE.

[Chen et al., 2021] Chen, H., Wang, Y., Guo, T., Xu, C., Deng, Y., Liu, Z., Ma, S., Xu, C., Xu, C., and Gao, W. (2021). Pre-trained image processing transformer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12299–12310.

[Chen et al., 2023] Chen, J., Yu, J., Ge, C., Yao, L., Xie, E., Wu, Y., Wang, Z., Kwok, J., Luo, P., Lu, H., and Li, Z. (2023). Pixart-α: Fast training of diffusion transformer for photorealistic text-to-image synthesis.

[Chu et al., 2018] Chu, M., Xie, Y., Leal-Taixé, L., and Thuerey, N. (2018). Temporally coherent gans for video super-resolution (tecogan). *arXiv preprint arXiv:1811.09393*, 1(2):3.

[Chu et al., 2020] Chu, M., Xie, Y., Mayer, J., Leal-Taixé, L., and Thuerey, N. (2020). Learning temporal coherence via self-supervision for gan-based video generation. *ACM Transactions on Graphics (TOG)*, 39(4):75–1.

[Chung et al., 2022] Chung, H., Kim, J., Mccann, M. T., Klasky, M. L., and Ye, J. C. (2022). Diffusion posterior sampling for general noisy inverse problems. *arXiv preprint arXiv:2209.14687*.

[Chung et al., 2023] Chung, H., Lee, S., and Ye, J. C. (2023). Decomposed diffusion sampler for accelerating large-scale inverse problems. *arXiv preprint arXiv:2303.05754*.

[Dai et al., 2007] Dai, S., Han, M., Xu, W., Wu, Y., and Gong, Y. (2007). Soft edge smoothness prior for alpha channel super resolution. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE.

[Daras et al., 2024] Daras, G., Nie, W., Kreis, K., Dimakis, A., Mardani, M., Kovachki, N., and Vahdat, A. (2024). Warped diffusion: Solving video inverse problems with image diffusion models. *Advances in Neural Information Processing Systems*, 37:101116–101143.

[Dhariwal and Nichol, 2021] Dhariwal, P. and Nichol, A. (2021). Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794.

[Dietterich, 2000] Dietterich, T. G. (2000). Ensemble methods in machine learning. In *International workshop on multiple classifier systems*, pages 1–15. Springer.

[Ding et al., 2021] Ding, K., Liu, Y., Zou, X., Wang, S., and Ma, K. (2021). Locally adaptive structure and texture similarity for image quality assessment. In *Proceedings of the 29th ACM International Conference on multimedia*, pages 2483–2491.

[Dong et al., 2014] Dong, C., Loy, C. C., He, K., and Tang, X. (2014). Learning a deep convolutional network for image super-resolution. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part IV 13*, pages 184–199. Springer.

[Dong et al., 2016] Dong, C., Loy, C. C., and Tang, X. (2016). Accelerating the super-resolution convolutional neural network. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14*, pages 391–407. Springer.

[Edge, 2025] Edge, T. A. (2025). The ai edge newsletter. Accessed: 2025-07-31.

[Efron, 2011] Efron, B. (2011). Tweedie's formula and selection bias. *Journal of the American Statistical Association*, 106(496):1602–1614.

[Farsiu et al., 2004] Farsiu, S., Robinson, M. D., Elad, M., and Milanfar, P. (2004). Fast and robust multiframe super resolution. *IEEE transactions on image processing*, 13(10):1327–1344.

[Freeman et al., 2002] Freeman, W. T., Jones, T. R., and Pasztor, E. C. (2002). Example-based super-resolution. *IEEE Computer graphics and Applications*, 22(2):56–65.

[Hénaff et al., 2019] Hénaff, O. J., Goris, R. L., and Simoncelli, E. P. (2019). Perceptual straightening of natural videos. *Nature neuroscience*, 22(6):984–991.

[Herzog and Clarke, 2014] Herzog, M. H. and Clarke, A. M. (2014). Why vision is not both hierarchical and feedforward. *Frontiers in computational neuroscience*, 8:135.

[Ho et al., 2020] Ho, J., Jain, A., and Abbeel, P. (2020). Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851.

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

[Hore and Ziou, 2010] Hore, A. and Ziou, D. (2010). Image quality metrics: Psnr vs. ssim. In *2010 20th international conference on pattern recognition*, pages 2366–2369. IEEE.

[Internò et al., 2025] Internò, C., Geirhos, R., Olhofer, M., Liu, S., Hammer, B., and Klindt, D. (2025). Ai-generated video detection via perceptual straightening. *arXiv preprint arXiv:2507.00583*.

[Kancharla and Channappayya, 2021a] Kancharla, P. and Channappayya, S. S. (2021a). Completely blind quality assessment of user generated video content. *IEEE Trans. on Image Processing*.

[Kancharla and Channappayya, 2021b] Kancharla, P. and Channappayya, S. S. (2021b). Completely blind quality assessment of user generated video content. *IEEE Transactions on Image Processing*, 30:4098–4111.

[Kancharla and Channappayya, 2021c] Kancharla, P. and Channappayya, S. S. (2021c). Improving the visual quality of video frame prediction models using the perceptual straightening hypothesis. *IEEE Signal Processing Letters*.

[Kancharla and Channappayya, 2021d] Kancharla, P. and Channappayya, S. S. (2021d). Improving the visual quality of video frame prediction models using the perceptual straightening hypothesis. *IEEE Signal Processing Letters*, 28:2167–2171.

[Kappeler et al., 2016] Kappeler, A., Yoo, S., Dai, Q., and Katsaggelos, A. K. (2016). Video super resolution with convolutional neural networks. *IEEE Trans. on Computational Imaging, vol. 2, no. 2*, 2(2):109–122.

[Keys, 1981] Keys, R. (1981). Cubic convolution interpolation for digital image processing. *IEEE transactions on acoustics, speech, and signal processing*, 29(6):1153–1160.

[Khachatryan et al., 2023] Khachatryan, L., Movsisyan, A., Tadevosyan, V., Henschel, R., Wang, Z., Navasardyan, S., and Shi, H. (2023). Text2video-zero: Text-to-image diffusion models are zero-shot video generators. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15954–15964.

[Kim et al., 2016] Kim, J., Lee, J. K., and Lee, K. M. (2016). Accurate image super-resolution using very deep convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1646–1654.

[Kim et al., 1990] Kim, S., Bose, N. K., and Valenzuela, H. M. (1990). Recursive reconstruction of high resolution image from noisy undersampled multiframes. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 38(6):1013–1027.

[Kirillova. et al., 2022] Kirillova., A., Lyapustin., E., Antsiferova., A., and Vatolin., D. (2022). Erqa: Edge-restoration quality assessment for video super-resolution. In *Proceedings of the 17th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - Volume 4: VISAPP,*, pages 315–322. INSTICC, SciTePress.

[Kwon and Ye, 2024] Kwon, T. and Ye, J. C. (2024). Vision-xl: High definition video inverse problem solver using latent image diffusion models. *arXiv preprint arXiv:2412.00156*.

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

[Li et al., 2022] Li, H., Yang, Y., Chang, M., Chen, S., Feng, H., Xu, Z., Li, Q., and Chen, Y. (2022). Srdiff: Single image super-resolution with diffusion probabilistic models. *Neurocomputing*, 479:47–59.

[Li and Orchard, 2001] Li, X. and Orchard, M. T. (2001). New edge-directed interpolation. *IEEE transactions on image processing*, 10(10):1521–1527.

[Liang et al., 2021] Liang, J., Cao, J., Sun, G., Zhang, K., Van Gool, L., and Timofte, R. (2021). Swinir: Image restoration using swin transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1833–1844.

[Lim et al., 2017] Lim, B., Son, S., Kim, H., Nah, S., and Mu Lee, K. (2017). Enhanced deep residual networks for single image super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 136–144.

[Liu et al., 2022] Liu, A., Liu, Y., Gu, J., Qiao, Y., and Dong, C. (2022). Blind image super-resolution: A survey and beyond. *IEEE transactions on pattern analysis and machine intelligence*, 45(5):5461–5480.

[Lu et al., 2021] Lu, Z., Liu, H., Li, J., and Zhang, L. (2021). Efficient transformer for single image super-resolution. *arXiv preprint arXiv:2108.11084*, 2.

[Lucas et al., 2019] Lucas, A., Lopez-Tapia, S., Molina, R., and Katsaggelos, A. K. (2019). Generative adversarial networks and perceptual losses for video super-resolution. *IEEE Trans. on Image Processing*, 28(7):3312–3327.

[Lugmayr et al., 2020] Lugmayr, A., Danelljan, M., Van Gool, L., and Timofte, R. (2020). Srflow: Learning the super-resolution space with normalizing flow. In *Computer vision–ECCV 2020: 16th European conference, glasgow, UK, August 23–28, 2020, proceedings, part v 16*, pages 715–732. Springer.

[Mittal et al., 2012] Mittal, A., Soundararajan, R., and Bovik, A. C. (2012). Making a "completely blind" image quality analyzer. *IEEE Signal processing letters*, 20(3):209–212.

[Nah et al., 2019] Nah, S. et al. (2019). NTIRE 2019 challenge on video super-resolution: Methods and results. In *IEEE Conf. on Comp. Vision and Pattern Recog. Workshops*.

[Pérez-Pellitero et al., 2018] Pérez-Pellitero, E., Sajjadi, M., Hirsch, M., and Schölkopf, B. (2018). Perceptual video super resolution with enhanced temporal consistency. *arXiv preprint arXiv:1807.07930*.

[Podell et al., 2023] Podell, D., English, Z., Lacey, K., Blattmann, A., Dockhorn, T., Müller, J., Penna, J., and Rombach, R. (2023). Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*.

[Pont-Tuset et al., 2017] Pont-Tuset, J., Perazzi, F., Caelles, S., Arbeláez, P., Sorkine-Hornung, A., and Van Gool, L. (2017). The 2017 davis challenge on video object segmentation. *arXiv:1704.00675*.

[Protter et al., 2008] Protter, M., Elad, M., Takeda, H., and Milanfar, P. (2008). Generalizing the nonlocal-means to super-resolution reconstruction. *IEEE Transactions on image processing*, 18(1):36–51.

[Qiao et al., 2019] Qiao, J., Song, H., Zhang, K., Zhang, X., and Liu, Q. (2019). Image super-resolution using conditional generative adversarial network. *IET Image Processing*, 13(14):2673–2679.

[Rahimi and Tekalp, 2023] Rahimi, N. and Tekalp, A. M. (2023). Spatio-temporal perception-distortion trade-off in learned video sr. In *2023 IEEE International Conference on Image Processing (ICIP)*, pages 1400–1404. IEEE.

[Robert and Casella, 2004] Robert, C. P. and Casella, G. (2004). *Monte Carlo Statistical Methods*. Springer.

[Rombach et al., 2022a] Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. (2022a). High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695.

[Rombach et al., 2022b] Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. (2022b). High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10684–10695.

[Rudin et al., 1992] Rudin, L. I., Osher, S., and Fatemi, E. (1992). Nonlinear total variation based noise removal algorithms. *Physica D: nonlinear phenomena*, 60(1-4):259–268.

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

[Seshadrinathan and Bovik, 2009] Seshadrinathan, K. and Bovik, A. C. (2009). Motion tuned spatio-temporal quality assessment of natural videos. *IEEE Trans. on image processing*, 19(2):335–350.

[Shi et al., 2016] Shi, W., Caballero, J., Huszár, F., Totz, J., Aitken, A. P., Bishop, R., Rueckert, D., and Wang, Z. (2016). Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1874–1883.

[Song et al., 2020] Song, J., Meng, C., and Ermon, S. (2020). Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*.

[Sun et al., 2018] Sun, D., Yang, X., Liu, M.-Y., and Kautz, J. (2018). Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8934–8943.

[Sun et al., 2008] Sun, J., Xu, Z., and Shum, H.-Y. (2008). Image super-resolution using gradient profile prior. In *2008 IEEE conference on computer vision and pattern recognition*, pages 1–8. IEEE.

[Takeda et al., 2009] Takeda, H., Milanfar, P., Protter, M., and Elad, M. (2009). Super-resolution without explicit subpixel motion estimation. *IEEE transactions on image processing*, 18(9):1958–1975.

[Tekalp et al., 2022] Tekalp, A. M. et al. (2022). Deep learning for image/video restoration and super-resolution. *Foundations and Trends® in Computer Graphics and Vision*, 13(1):1–110.

[Tihonov and Arsenin, 1977] Tihonov, N. and Arsenin, V. (1977). Solution of ill-posed problems: Winston. *Washington, DC*.

[Tong et al., 2017] Tong, T., Li, G., Liu, X., and Gao, Q. (2017). Image super-resolution using dense skip connections. In *Proceedings of the IEEE international conference on computer vision*, pages 4799–4807.

[Tsai and Huang, 1984] Tsai, R. Y. and Huang, T. S. (1984). Multiframe image restoration and registration. *Multiframe image restoration and registration*, 1:317–339.

[Tu et al., 2020] Tu, Z., Chen, C.-J., Chen, L.-H., Birkbeck, N., Adsumilli, B., and Bovik, A. C. (2020). A comparative evaluation of temporal pooling methods for blind video quality assessment. In *IEEE Int. Conf. on Image Processing (ICIP)*, pages 141–145.

[Unterthiner et al., 2019] Unterthiner, T., Van Steenkiste, S., Kurach, K., Marinier, R., Michalski, M., and Gelly, S. (2019). Fvd: A new metric for video generation.

[Vandewalle et al., 2007] Vandewalle, P., Krichane, K., Alleysson, D., and Süsstrunk, S. (2007). Joint demosaicing and super-resolution imaging from a set of unregistered aliased images. In *Digital Photography III*, volume 6502, pages 78–89. SPIE.

[Vaswani et al., 2017] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30.

[Wang et al., 2019] Wang, X., Chan, K. C., Yu, K., Dong, C., and Change Loy, C. (2019). Edvr: Video restoration with enhanced deformable convolutional networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 0–0.

[Wang et al., 2018] Wang, X., Yu, K., Wu, S., Gu, J., Liu, Y., Dong, C., Qiao, Y., and Change Loy, C. (2018). Esrgan: Enhanced super-resolution generative adversarial networks. In *Proceedings of the European conference on computer vision (ECCV) workshops*, pages 0–0.

[Wang et al., 2004] Wang, Z., Bovik, A. C., Sheikh, H. R., and Simoncelli, E. P. (2004). Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612.

[Xie et al., 2018] Xie, Y., Franz, E., Chu, M., and Thuerey, N. (2018). Tempogan: A temporally coherent, volumetric gan for super-resolution fluid flow. *ACM Trans. on Graphics (TOG)*, 37(4):1–15.

[Yeh et al., 2024] Yeh, C.-H., Lin, C.-Y., Wang, Z., Hsiao, C.-W., Chen, T.-H., Shiu, H.-S., and Liu, Y.-L. (2024). Diffir2vr-zero: Zero-shot video restoration with diffusion-based image restoration models. *arXiv preprint arXiv:2407.01519*.

[Zhang et al., 2018a] Zhang, R., Isola, P., Efros, A. A., Shechtman, E., and Wang, O. (2018a). The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595.

[Zhang et al., 2018b] Zhang, Y., Tian, Y., Kong, Y., Zhong, B., and Fu, Y. (2018b). Residual dense network for image super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2472–2481.

[Zhao and Sawhney, 2002] Zhao, W. and Sawhney, H. S. (2002). Is super-resolution with optical flow feasible? In *European Conference on Computer Vision*, pages 599–613. Springer.

[Zhou et al., 2024] Zhou, S., Yang, P., Wang, J., Luo, Y., and Loy, C. C. (2024). 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*, pages 2535–2545.