DINTR: Tracking via Diffusion-based Interpolation

Pha Nguyen[1], **Ngan Le**[1], **Jackson Cothren**[1], **Alper Yilmaz**[2], **Khoa Luu**[1]
[1]University of Arkansas [2]Ohio State University
[1]{panguyen, thile, jcothre, khoaluu}@uark.edu [2]yilmaz.15@osu.edu

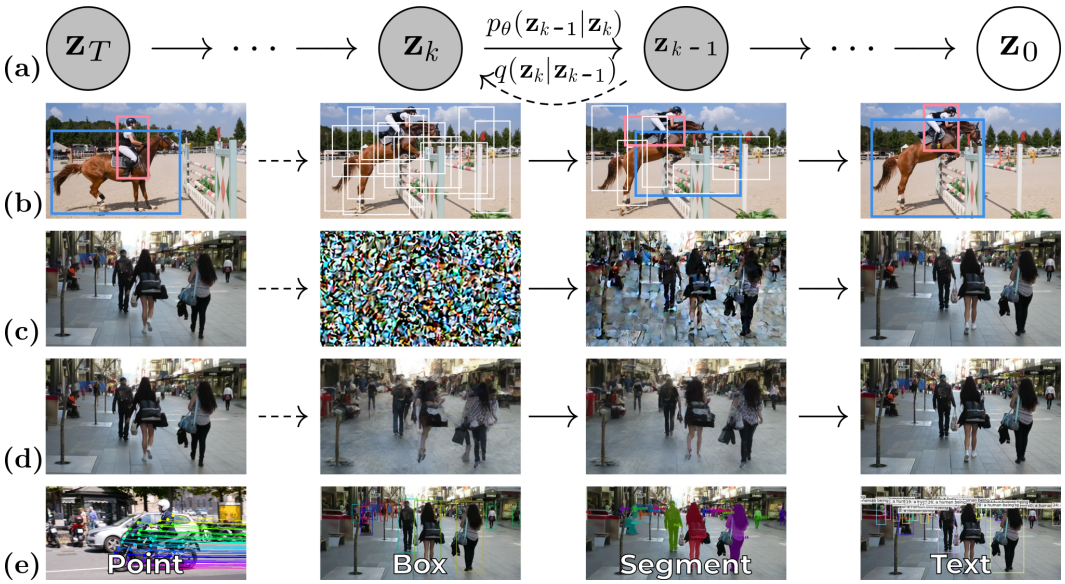


Figure 1: Diffusion-based processes. (a) Probabilistic diffusion process [1], where $q(\cdot)$ is noise sampling and $p_\theta(\cdot)$ is denoising. (b) Diffusion process in the 2D coordinate space [2, 3, 4]. (c) A purely visual diffusion-based *data prediction* approach reconstructs the subsequent video frame. (d) Our proposed *data interpolation* approach **DINTR** interpolates between two consecutive video frames, indexed by timestamp t, allowing a seamless temporal transition for visual content understanding, temporal modeling, and instance extracting for the object tracking task across various indications (e).

Abstract

Object tracking is a fundamental task in computer vision, requiring the localization of objects of interest across video frames. Diffusion models have shown remarkable capabilities in visual generation, making them well-suited for addressing several requirements of the tracking problem. This work proposes a novel diffusion-based methodology to formulate the tracking task. Firstly, their conditional process allows for injecting indications of the target object into the generation process. Secondly, diffusion mechanics can be developed to inherently model temporal correspondences, enabling the reconstruction of actual frames in video. However, existing diffusion models rely on extensive and unnecessary mapping to a Gaussian noise domain, which can be replaced by a more efficient and stable interpolation process. Our proposed interpolation mechanism draws inspiration from classic image-processing techniques, offering a more interpretable, stable, and faster approach tailored specifically for the object tracking task. By leveraging the strengths of diffusion models while circumventing their limitations, our **D**iffusion-based **IN**terpolation Tracke**R** (**DINTR**) presents a promising new paradigm and achieves a superior multiplicity on seven benchmarks across five indicator representations.

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

https://doi.org/10.52202/079017-2221

1 Introduction

Object tracking is a long-standing computer vision task with widespread applications in video analysis and instance-based understanding. Over the past decades, numerous tracking paradigms have been explored, including *tracking-by-regression* [5], *-detection* [6], *-segmentation* [7] and two more recent *tracking-by-attention* [8, 9], *-unification* [10] paradigms. Recently, generative modeling has achieved great success, offering several promising new perspectives in instance recognition. These include denoising sampling bounding boxes to final prediction [2, 3, 4], or sampling future trajectories [11]. Although these studies explore the generative process in instance-based understanding tasks, they perform solely on coordinate refinement rather than performing on the visual domain, as in Fig. 1b.

In this work, we propose a novel tracking framework solely based on *visual* iterative latent variables of diffusion models [12, 13], thereby introducing the novel and true *Tracking-by-Diffusion* paradigm. This paradigm demonstrates versatile applications across various indications, comprising points, bounding boxes, segments, and textual prompts, facilitated by the conditional mechanism (Eqn. (3)).

Moreover, our proposed **D**iffusion-based **IN**terpolation **T**racke**R** (**DINTR**) inherently models the temporal correspondences via the diffusion mechanics, *i.e*., the denoising process. Specifically, by formulating the process to operate temporal modeling *online* and *auto-regressively* (*i.e*. next-frame reconstruction, as in Eqn. (4)), **DINTR** enables the capability for instance-based video understanding tasks, specifically the object tracking. However, existing diffusion mechanics rely on an extensive and unnecessary mapping to a Gaussian noise domain, which we argue can be replaced by a more efficient interpolation process (Subsection 4.3). Our proposed interpolation operator draws inspiration from the image processing field, offering a more direct, seamless, and stable approach. By leveraging the diffusion mechanics while circumventing their limitations, our **DINTR** achieves superior multiplicity on seven benchmarks across five types of indication, as elaborated in Section 5. Note that our Interpolation process does not aim to generate high-fidelity unseen frames [14, 15, 16, 17]. Instead, its objective is to seamlessly transfer internal states between frames for visual semantic understanding.

Contributions. Overall, ***(i)*** this paper reformulates the *Tracking-by-Diffusion* paradigm to operate on visual domain ***(ii)*** which demonstrates broader tracking applications than existing paradigms. ***(iii)*** We reformulate the diffusion mechanics to achieve two goals, including *(a)* temporal modeling and *(b)* iterative interpolation as a $2\times$ faster process. ***(iv)*** Our proposed **DINTR** achieves superior multiplicity and State-of-the-Art (SOTA) performances on *seven tracking benchmarks* of *five representations*. ***(v)*** Following sections including **Appendices** A elaborate on its formulations, properties, and evaluations.

2 Related Work

2.1 Object Tracking Paradigms

Tracking-by-Regression methods refine future object positions directly based on visual features. Previous approaches [31, 45] rely on the regression branch of object features in nearby regions. CenterTrack [5] represents objects via center points and temporal offsets. It lacks explicit object identity, requiring the appearance [31], motion model [46], and graph matching [47] components.

Tracking-by-Detection methods form object trajectories by linking detections over consecutive frames, treating the task as an optimization problem. *Graph*-based methods formulate the tracking problem as a bipartite matching or maximum flow [48]. These methods utilize a variety of techniques, such as link prediction [49], trainable graph neural networks [47, 34], edge lifting [50], weighted graph labeling [51], multi-cuts [52, 53], general-purpose solvers [54], motion information [55], learned models [56], association graphs [57], and distance-based [58, 59, 60]. Additionally, *Appearance*-based methods leverage robust image recognition frameworks to track objects. These techniques depend on similarity measures derived from 3D appearance and pose [61], affinity estimation [62], detection candidate selection [62], learned re-identification features [63, 64], or twin neural networks [65]. On the other hand, *Motion* modeling is leveraged for camera motion [66], observation-centric manner [67], trajectory forecasting [11], the social force model [68, 69, 70, 71], based on constant velocity assumptions [72, 73], or location estimation [74, 68, 75] directly from trajectory sequences. Additionally, data-driven motion [76] need to project 3D into 2D motions [77].

Tracking-by-Segmentation leverages detailed pixel information and addresses the challenges of unclear backgrounds and crowded scenes. Methods include cost volumes [7], point cloud representa-

https://doi.org/10.52202/079017-2221

Table 1: Comparison of paradigms, mechanisms of SOTA tracking methods. **Indication Types** defines the representation to indicate targets with their corresponding datasets: **TAP-Vid** [18], **PoseTrack** [19, 20], **MOT** [21, 22, 23], **VOS** [24], **VIS** [25], **MOTS** [26], **KITTI** [27], **LaSOT** [28], **GroOT** [29]. **Methods** in color gradient support both types of **single-** and **multi-**target benchmarks.

Method	Paradigm	Mechanism*	Indication Types				
			Point	Pose	Box	Segment	Text
TAPIR [30]	Regression	*Iter.* Refinement	**TAP-Vid**	✗	✗	✗	✗
Tracktor++ [31]		Regression Head	✗	✗	**MOT**	✗	✗
CenterTrack [5]		Offset Prediction	✗	✗	**MOT**	✗	✗
GTI [32]		*Rgn-Tpl Integ.*	✗	✗	**LaSOT**	✗	**LaSOT**
DeepSORT [33]	Detection	Cascade *Assoc.*	✗	✗	**MOT**	✗	✗
GSDT [34]		Relation Graph	✗	✗	**MOT**	✗	✗
JDE [35]		Multi-Task	✗	✗	**MOT**	✗	✗
ByteTrack [36]		Two-stage *Assoc.*	✗	✗	**MOT**	✗	✗
TrackR-CNN [37]	Segmentation	3D Convolution	✗	✗	✗	**MOTS**	✗
MOTSNet [26]		Mask-Pooling	✗	✗	✗	**MOTS**	✗
CAMOT [38]		Hypothesis Select	✗	✗	✗	**KITTI**	✗
PointTrack [39]		*Seg.* as Points	✗	✗	✗	**MOTS/KITTI**	✗
MixFormerV2 [40]	Attention	Mixed Attention	✗	✗	**LaSOT**	✗	✗
TransVLT [41]		*X*-Modal Fusion	✗	✗	**LaSOT**	✗	**LaSOT**
MeMOTR [42]		Memory *Aug.*	✗	✗	**MOT**	✗	✗
MENDER [29]		Tensor *Decomp.*	✗	✗	**MOT**	✗	**GroOT**
SiamMask [43]	Unification	Variant Head	✗	✗	**LaSOT**	**VOS**	✗
TraDeS [7]		Cost Volume	✗	✗	**MOT**	**VIS/MOTS**	✗
UNICORN [10]		Unified *Embed.*	✗	✗	**LaSOT/MOT**	**VOS/MOTS**	✗
UniTrack [44]		Primitive Level	✗	**PoseTrack**	**LaSOT/MOT**	**VOS/MOTS**	✗
DiffusionTrack [3]	**Diffusion**	Denoised *Coord.*	✗	✗	**MOT**	✗	✗
DiffMOT [4]		*Motion* Predictor	✗	✗	**MOT**	✗	✗
DINTR (Ours)		Visual *Interpolat.*	**TAP-Vid**	**PoseTrack**	**LaSOT/MOT**	**VOS/MOTS**	**LaSOT/GroOT**

* *Iter.*: Iterative. *Rgn-Tpl Integ.*: Region-Template Integration. *Assoc.*: Association. *X*: Cross. *Decomp.*: Decomposition. *Embed.*: Embedding. *Coord.*: **2D** Coordinate. *Motion*: **2D** Motion. *Interpolat.*: Interpolation.

tions [39], mask pooling layers [26], and mask-based [38] with 3D convolutions [37]. However, its reliance on segmented multiple object tracking data often necessitates bounding box initialization.

Tracking-by-Attention applies the attention mechanism [78] to link detections with tracks at the feature level, represented as tokens. TrackFormer [8] approaches tracking as a unified prediction task using attention, during initiation. MOTR [9] and MOTRv2 [79] advance this concept by integrating motion and appearance models, aiding in managing object entrances/exits and temporal relations. Furthermore, object token representations can be enhanced via memory techniques, such as memory augmentation [42] and memory buffer [80, 81]. Recently, MENDER [29] presents another stride, a transformer architecture with tensor decomposition to facilitate object tracking through descriptions.

Tracking-by-Unification aims to develop unified frameworks that can handle multiple tasks simultaneously. Pioneering works in this area include TraDeS [7] and SiamMask [43], which combine object tracking (SOT/MOT) and video segmentation (VOS/VIS). UniTrack [44] employs separate task-specific heads, enabling both object propagation and association across frames. Furthermore, UNICORN [10] investigates learning robust representations by consolidating from diverse datasets.

2.2 Diffusion Model in Semantic Understanding

Generative models have recently been found to be capable of performing understanding tasks.

Visual Representation and Correspondence. Hedlin *et al*. [82] establishes semantic visual correspondences by optimizing text embeddings to focus on specific regions. Diffusion Autoencoders [83] form a diffusion-based autoencoder encapsulating high-level semantic information. Similarly, Zhang *et al*. [84] combine features from Stable Diffusion (SD) and DINOv2 [85] models, effectively merging the high-quality spatial information and capitalizing on both strengths. Diffusion Hyperfeatures [86] uses feature aggregation and transforms intermediate feature maps from the diffusion process into a single, coherent descriptor map. Concurrently, DIFT [87] simulates the forward diffusion process, adding noise to input images and extracting features within the U-Net. Asyrp [88] employs the asymmetric reverse process to explore and manipulate a semantic latent space, upholding the original

https://doi.org/10.52202/079017-2221

performance, integrity, and consistency. Furthermore, DRL [89] introduces an infinite-dimensional latent code that offers discretionary control over the granularity of detail.

Generative Perspectives in Object Tracking. A straightforward application of generative models in object tracking is to augment and enrich training data [90, 91, 92]. For trajectory refinement, QuoVadis [11] uses the social generative adversarial network (GAN) [93] to sample future trajectories to account for the uncertainty in future positions. DiffusionTrack [3] and DiffMOT [4] utilize the diffusion process in the bounding box decoder. Specifically, they pad prior *2D coordinate* bounding boxes with noise, then transform them into tracking results via a denoising decoder.

2.3 Discussion

This subsection discusses the key aspects of our proposed paradigm and method, including the mechanism comparison of our **DINTR** against alternative diffusion approaches [2, 3, 4], and the properties that enable *Tracking-by-Diffusion* on visual domain to stand out from the existing paradigms.

Conditioning Mechanism. As illustrated in Fig. 1b, tracking methods performing diffusion on the 2D coordinate space [3, 4] utilize generative models to model 2D object motion or refine coordinate predictions. However, they fail to leverage the conditioning mechanism [13] of Latent Diffusion Models, which are principally capable of modeling unified conditional distributions. As a result, these diffusion-based approaches have a specified indicator representation limited to the bounding box, that cannot be expanded to other advanced indications, such as point, pose, segment, and text.

In contrast, we formulate the object tracking task as two visual processes, including one for diffusion-based Reconstruction, as illustrated in Fig. 1c, and another $2\times$ faster approach that is Interpolation, as shown in Fig. 1d. These two approaches demonstrate their superior versatility due to the controlled injection $p_\theta(\mathbf{z}|\tau)$ implemented by the attention mechanism [78] (Eqn. (3)) during iterative diffusion.

Unification. Current methods under *tracking-by-unification* face challenges due to the separation of task-specific heads. This issue arises because single-object and multi-object tracking tasks are trained on distinct branches [7, 44] or stages [35], with results produced through a manually designed decoder for each task. The architectural discrepancies limit the full utilization of network capacity.

In contrast, *Tracking-by-Diffusion* operating on the visual domain addresses the limitations of unification. Our method seamlessly handles diverse tracking objectives, including *(a) point and pose* ***regression***, *(b) bounding box and segmentation* ***prediction***, and *(c) referring* ***initialization***, while remaining *(d)* ***data-*** *and* ***process-unified*** through an iterative process. This is possible because our approach operates on the base core domain, allowing it to understand contexts and extract predictions.

Application Coverage presented in Table 1 validates the unification advantages of our approach. As highlighted, our proposed model **DINTR** supports unified tracking across *seven benchmarks* of *eight settings* comprising *five distinct categories of indication*. It can handle both **single-target** and **multiple-target** benchmarks, setting a new standard in terms of multiplicity, flexibility, and novelty.

3 Problem Formulation

Given two images $\mathbf{I}_t$ and $\mathbf{I}_{t+1}$ from a video sequence $\mathcal{V}$, and an indicator representation L_t (*e.g.*, point, structured points set for pose, bounding box, segment, or text) for an object in $\mathbf{I}_t$, our goal is to find the respective region L_{t+1} in $\mathbf{I}_{t+1}$. The relationship between L_t and L_{t+1} can encode semantic correspondences [87, 86, 94] (*i.e.*, different objects with similar semantic meanings), geometric correspondence [95, 96, 97] (*i.e.*, the same object viewed from different viewpoints) or temporal correspondence [98, 99, 100] (*i.e.*, the location of a deforming object over a video sequence).

We define the object-tracking task as temporal correspondence, aiming to establish matches between regions representing the same real-world object as it moves, potentially deforming or occluding across the video sequence over time. Let us denote a feature encoder $\mathcal{E}(\cdot)$ that takes as input the frame $\mathbf{I}_t$ and returns the feature representation $\mathbf{z}^t$. Along with the region L_t for the initial indication, the ***online*** *and* ***auto-regressive*** *objective* for the tracking task can be written as follows:

$$L_{t+1} = \arg\min_{L} dist\big(\mathcal{E}(\mathbf{I}_t)[L_t], \mathcal{E}(\mathbf{I}_{t+1})[L]\big), \tag{1}$$

where $dist(\cdot,\cdot)$ is a semantic distance that can be cosine [33] or distributional softmax [101]. A special case is to give L_t as textual input and return L_{t+1} as a bounding box for the *referring object*

tracking [29, 102] task. In addition, the pose is treated as multiple-point tracking. The output L_{t+1} is then mapped to a point, box, or segment. We explore how diffusion models can learn these temporal dynamics end-to-end to output consistent object representations frame-to-frame in the next section.

4 Methodology

This section first presents the notations and background. Then, we present the deterministic frame reconstruction task for video modeling. Finally, our proposed framework **DINTR** is introduced.

4.1 Notations and Background

Latent Diffusion Models (LDMs) [1, 13, 103] are introduced to denoise the latent space of an autoencoder. First, the encoder $\mathcal{E}(\cdot)$ compresses a RGB image $\mathbf{I}_t$ into an initial latent space $\mathbf{z}_0^t = \mathcal{E}(\mathbf{I}_t)$, which can be reconstructed to a new image $\mathcal{D}(\mathbf{z}_0^t)$. Let us denote two operators $\mathcal{Q}$ and $\mathcal{P}_{\varepsilon_\theta}$ are corresponding to the sampling noise process $q(\mathbf{z}_k^t|\mathbf{z}_{k-1}^t)$ and the denoising process $p_\varepsilon(\mathbf{z}_{k-1}^t|\mathbf{z}_k^t)$, where $\mathcal{P}_{\varepsilon_\theta}$ is parameterized by an U-Net ε_θ [104] as a *noise prediction model* via the objective:

$$\min_\theta \mathbb{E}_{\mathbf{z}_0^t, \epsilon\sim\mathcal{N}(0,1), k\sim\mathcal{U}(1,T)} \Big[\big\| \epsilon - \mathcal{P}_{\varepsilon_\theta}\big(\mathcal{Q}(\mathbf{z}_0^t, k), k, \tau\big) \big\|_2^2 \Big], \qquad \text{where } \tau = \mathcal{T}_\theta(L_t). \tag{2}$$

Localization. All types of localization L_t, *e.g.*, point, pose (*i.e.* set of structured points), bounding box, segment, and especially text, are unified as guided indicators. $\mathcal{T}_\theta(\cdot)$ is the respective extractor, such as the Gaussian kernel for point, pooling layer for bounding box and segment, or word embedding model for text. $\mathbf{z}_k^t$ is a noisy sample of $\mathbf{z}_0^t$ at step $k \in [1, \dots, T]$, and $T = 50$ is the maximum step.

The Conditional Process $p_\theta(\mathbf{z}_0^{t+1}|\tau)$, containing cross-attention $Attn(\varepsilon, \tau)$ to inject the indication τ to an autoencoder with U-Net blocks $\varepsilon_\theta(\cdot,\cdot)$, is derived after noise sampling $\mathbf{z}_k^t = \mathcal{Q}(\mathbf{z}_0^t, k)$:

$$\mathcal{P}_{\varepsilon_\theta}\big(\mathcal{Q}(\mathbf{z}_0^t, k), k, \tau\big) = \underbrace{\text{softmax}\Big(\frac{\varepsilon_\theta(\overbrace{\sqrt{\bar\alpha_k}\mathbf{z}_0^t + \sqrt{1-\bar\alpha_k}\epsilon}^{\mathcal{Q}(\mathbf{z}_0^t,k)}) \times W_Q \times (\tau \times W_K)^\intercal}{\sqrt{d}}\Big)}_{Attn(\varepsilon,\tau)} \times (\tau \times W_V), \tag{3}$$

where $W_{Q,K,V}$ are projection matrices, d is the feature size, and α_k is a scheduling parameter.

4.2 Deterministic Next-Frame Reconstruction by Data Prediction Model

The *noise prediction model*, defined in Eqn. (2), can not generate specific desired pixel content while denoising the latent feature to the new image. To effectively model and generate exactly the desired video content, we formulate a next-frame reconstruction task, such that $\mathcal{D}(\mathcal{P}_{\varepsilon_\theta}(\mathbf{z}_T^t, T, \tau)) \approx \mathbf{I}_{t+1}$. In this formulation, the denoised image obtained from the diffusion process should approximate the next frame in the video sequence. The objective for a *data prediction model* (Fig. 1c) derives that goal as:

$$\min_\theta \mathbb{E}_{\mathbf{z}_0^{t,t+1}, k\sim\mathcal{U}(1,T)} \Big[\big\| \mathbf{z}_k^{t+1} - \mathcal{P}_{\varepsilon_\theta}\big(\mathcal{Q}(\mathbf{z}_0^t, k), k, \tau\big) \big\|_2^2 \Big]. \tag{4}$$

In layman's terms, the objective of the *data prediction model* formulates the task of establishing temporal correspondence between frames by effectively capturing the pixel-level changes and reconstructing the real next frame from the current frame. With the pre-trained decoder $\mathcal{D}(\cdot)$ in place, the key optimization target becomes the denoising process itself. To achieve this, a combination of step-wise KL divergences is used to guide the likelihood of current frame latents $\mathbf{z}_k^t$ toward the desired latent representations for the next frame $\mathbf{z}_k^{t+1}$, as described in Alg. 1 and derived as:

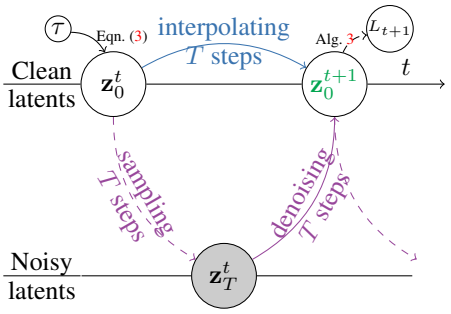
Algorithm 1 Inplace Reconstruction Finetuning

Input: Network ε_θ, video sequence $\mathcal{V}$, indication $L_{t=0}$

Sample $(t, t+1) \sim \mathcal{U}(0, |\mathcal{V}| - 2)$
  $\tau \leftarrow \mathcal{T}_\theta(L_t)$
  Draw $\mathbf{I}_{t,t+1} \in \mathcal{V}$ and encode $\mathbf{z}_0^{t,t+1} = \mathcal{E}(\mathbf{I}_{t,t+1})$
  Sample $k \sim \mathcal{U}(1, T)$
    Optimize $\min_\theta \big[\|\mathbf{z}_k^{t+1} - \mathcal{P}_{\varepsilon_\theta}(\mathcal{Q}(\mathbf{z}_0^t, k), k, \tau)\|_2^2\big]$
  Optimize $\min_\theta \big[\|\mathbf{I}_{t+1} - \mathcal{D}(\mathcal{P}_{\varepsilon_\theta}(\mathcal{Q}(\mathbf{z}_0^t, k), k, \tau))\|_2^2\big]$

$$\mathcal{L} = \frac{1}{2}\mathbb{E}_{\mathbf{z}_0^{t,t+1}, k\sim\mathcal{U}(1,T)} \big[\|\mathbf{z}_k^{t+1} - \mathcal{P}_{\varepsilon_\theta}(\mathcal{Q}(\mathbf{z}_0^t, k), k, \tau)\|_2^2\big] = \int_0^1 \frac{d}{d\alpha_k} D_{KL}\big(q(\mathbf{z}_k^{t+1}|\mathbf{z}_{k-1}^{t+1}) \| p_\varepsilon(\mathbf{z}_{k-1}^t|\mathbf{z}_k^t)\big)\, d\alpha_k. \tag{5}$$

https://doi.org/10.52202/079017-2221



Algorithm 2 Temporal Interpolation in **DINTR**

Input: Network ϕ_θ, latent feature $\mathbf{z}_0^t$, $\tau \leftarrow \mathcal{T}_\theta(L_0)$

1: Initialize $\widehat{\mathbf{z}}_T^{t+1} \leftarrow \mathbf{z}_0^t$
2: **for** $k \in \{T, \dots, 0\}$ **do**
3: $\quad \widehat{\mathbf{z}}_k^{t+1} \leftarrow \mathcal{P}_{\phi_\theta}(\widehat{\mathbf{z}}_k^{t+1}, k, \tau)$; **if** $k = 0$ **then** break
4: $\quad \widehat{\mathbf{z}}_{k-1}^{t+1} \leftarrow \widehat{\mathbf{z}}_k^{t+1} - \mathcal{Q}(\mathbf{z}_0^t, k) + \mathcal{Q}(\mathbf{z}_0^{t+1}, k-1)$
5: **end for**
6: **return** $\{\widehat{\mathbf{z}}_k^{t+1} \mid k \in \{T, \dots, 0\}\}$

Figure 2: Illustration of the reconstruction and interpolation processes, where the purple dashed arrow is $q(\mathbf{z}_T^t|\mathbf{z}_0^t)$ and the purple solid arrow is $p_\varepsilon(\mathbf{z}_0^{t+1}|\mathbf{z}_T^t)$, while the blue arrow illustrates $p_\phi(\mathbf{z}_0^{t+1}|\mathbf{z}_0^t)$.

where $\alpha_k = \frac{k}{T}$. This loss function constructed from the extensive step-wise divergences creates an accumulative path between the visual distributions. Instead, we propose to employ the classic interpolation operator used in image processing to formulate a new diffusion-based process that iteratively learns to blend video frames. This interpolation approach ultimately converges towards the same deterministic mapping toward $\mathbf{z}_0^{t+1}$ but is simpler to derive and more stable. The proposed process is illustrated in Fig. 2, and interpolation operators are elaborated in the next Subsection 4.3.

4.3 DINTR for Tracking via Diffusion-based Interpolation

Denoising Process as Temporal Interpolation. We relax the controlled Gaussian space projection of every step. Specifically, we impose a temporal bias by training a *data interpolation model* ϕ_θ. The data interpolation process is denoted as $\mathcal{P}_{\phi_\theta}$ producing intermediate interpolated features $\widehat{\mathbf{z}}_k^{t+1}$, so that $\mathcal{P}_{\phi_\theta}(\mathbf{z}_0^t, T, \tau) = \widehat{\mathbf{z}}_0^{t+1} \approx \mathbf{z}_0^{t+1}$. The goal is to obtain $p_\phi(\mathbf{z}_0^{t+1}|\mathbf{z}_0^t)$ by optimizing the objective:

$$\min_\theta \mathbb{E}_{\mathbf{z}_0^{t,t+1}} \left[\|\mathbf{z}_0^{t+1} - \mathcal{P}_{\phi_\theta}(\mathbf{z}_0^t, T, \tau)\|_2^2 \right]. \tag{6}$$

This *data interpolation model* ϕ_θ (Fig. 1d) allows us to derive a straightforward *single-step* loss as:

$$\mathcal{L} = D_{KL}\big(\mathbf{z}_0^{t+1} \parallel p_\phi(\mathbf{z}_0^{t+1}|\mathbf{z}_0^t)\big) = \log \frac{\mathbf{z}_0^{t+1}}{p_\phi(\mathbf{z}_0^{t+1}|\mathbf{z}_0^t)}. \tag{7}$$

The simplicity of the loss function comes from the knowledge that we are directly modeling the frame transition in the latent space, that is, $\widehat{\mathbf{z}}_k^{t+1} \approx \mathbf{z}_k^{t+1}$ where $k \in \{T, \dots, 1\}$ is not required. Therefore, we do not use the noise sampling operator $\mathcal{Q}(\cdot)$ as in the step-wise reconstruction objective defined in Eqn. (4). Instead, noise is added in the form of an offset, as described in L4 of Alg. 2. Note that the same network structure of ε_θ can be used for ϕ_θ without changing layers. Additionally, with the base case $\widehat{\mathbf{z}}_T^{t+1} = \mathbf{z}_0^t$, the transition is *accumulative* within the *inductive* data interpolation itself:

$$\begin{aligned} &k \in \{T-1, \dots, 1\}, \\ &\Big(\underbrace{\mathcal{P}_{\phi_\theta}\big(\widehat{\mathbf{z}}_{k+1}^{t+1} + (\mathbf{z}_k^{t+1} - \mathbf{z}_{k+1}^t), k, \tau\big)}_{\widehat{\mathbf{z}}_k^{t+1}} \rightarrow \mathcal{P}_{\phi_\theta}\big(\widehat{\mathbf{z}}_k^{t+1} \underbrace{+(\mathbf{z}_{k-1}^{t+1} - \mathbf{z}_k^t)}_{\text{Interpolation operator}}, k-1, \tau\big) \Big). \end{aligned} \tag{8}$$

Table 2: Equivalent formulation of interpolative operators, where $\mathbf{z}_{k,k-1}^{t,t+1} = \mathcal{Q}\big(\mathbf{z}_0^{t,t+1}, [k, k-1]\big)$.

(a) **linear blending**	(b) **learning from** $\mathbf{z}_0^{t+1}$	(c) **learning from** $\mathbf{z}_0^t$	(d) **learning offset**
$\widehat{\mathbf{z}}_{k-1}^{t+1} = \alpha_{k-1}\,\mathbf{z}_0^t + (1-\alpha_{k-1})\,\mathbf{z}_0^{t+1}$	$\widehat{\mathbf{z}}_{k-1}^{t+1} = \mathbf{z}_0^{t+1} + \frac{\alpha_{k-1}}{\alpha_k}(\widehat{\mathbf{z}}_k^{t+1} - \mathbf{z}_0^{t+1})$	$\widehat{\mathbf{z}}_{k-1}^{t+1} = \mathbf{z}_0^t + \frac{1-\alpha_{k-1}}{1-\alpha_k}(\widehat{\mathbf{z}}_k^{t+1} - \mathbf{z}_0^t)$	$\widehat{\mathbf{z}}_{k-1}^{t+1} = \widehat{\mathbf{z}}_k^{t+1} + (\alpha_k - \alpha_{k-1})(\mathbf{z}_{k-1}^{t+1} - \mathbf{z}_k^t)$
stable	*unstable*, when $\alpha_k \to 0$	*unstable*, when $\alpha_k \to 1$	*stable*
deterministic	*nondeterm.*, missing $\mathbf{z}^t$	*nondeterm.*, missing $\mathbf{z}^{t+1}$	*deterministic*
nonaccumulative	*accumulative*, Eqn. (C.19)	*accumulative*, Eqn. (C.21)	*accumulative*, Eqn. (8)

https://doi.org/10.52202/079017-2221

Interpolation Operator is selected based on the theoretical properties between the equivalent variants [105], presented in Table 2 and derived in Section C. In this table, we define $\alpha_k = \frac{k}{T}$, then the selected operator (2d), which adds noise in offset form $\mathcal{Q}(\mathbf{z}_0^{t+1}, k-1) - \mathcal{Q}(\mathbf{z}_0^t, k)$, is derived as:

$$\widehat{\mathbf{z}}_{k-1}^{t+1} = \widehat{\mathbf{z}}_k^{t+1} + (\alpha_k - \alpha_{k-1})\,(\mathbf{z}_{k-1}^{t+1} - \mathbf{z}_k^t) = \widehat{\mathbf{z}}_k^{t+1} + \frac{k-(k-1)}{T}\,(\mathbf{z}_{k-1}^{t+1} - \mathbf{z}_k^t), \tag{9}$$

$$\propto \widehat{\mathbf{z}}_k^{t+1} + (\mathbf{z}_{k-1}^{t+1} - \mathbf{z}_k^t) = \widehat{\mathbf{z}}_k^{t+1} - \mathcal{Q}(\mathbf{z}_0^t, k) + \mathcal{Q}(\mathbf{z}_0^{t+1}, k-1), \qquad \text{as in L4 of Alg. 2.} \tag{10}$$

Intuitively, the proposed interpolation process to generate the next frame takes the current frame as the starting point of the noisy sample. The internal states and intermediate features of the diffusion model transition from the current frame, resulting in a more stable prediction for video modeling.

Correspondence Extraction via Internal States. From Eqn. (3), we demonstrate that *the object of interest can be injected via the indication*. From the objectives in Eqn. (4) and Eqn. (6), we show that *the next frame* $\mathbf{I}_{t+1}$ *can be* ***reconstructed*** *or* ***interpolated*** *from the current frame* $\mathbf{I}_t$. Subsequently, internal accumulative and stable states, such as the attention map $Attn(\cdot,\cdot)$, which exhibit spatial correlations, can be used to identify the target locations and can be effortlessly extracted. To get into that, the self- and cross-attention maps $(\bar{\mathcal{A}}_S, \bar{\mathcal{A}}_X)$ over N layers and T time steps are averaged and performed element-wise multiplication:

Algorithm 3 Correspondence Extraction

Input: Internal $Attn$'s while processing $\mathcal{P}_{\phi_\theta}$

for $k \in [0, T \times 0.8]$ **do**
  $\mathcal{A}_{S,X}$ += $\sum_{l=1}^{N} \left[Attn_{[l,k]}(\varepsilon,\varepsilon), Attn_{[l,k]}(\varepsilon,\tau)\right]$
end for ▷ requires *accumulativeness* in Table 2
$\bar{\mathcal{A}}_{S,X} \leftarrow \frac{1}{N\times T\times 0.8} \sum_{k=0}^{T\times 0.8} \mathcal{A}_{S,X}$
$\bar{\mathcal{A}}^* \leftarrow \bar{\mathcal{A}}_S \circ \bar{\mathcal{A}}_X$
$L_{t+1} \leftarrow \text{map}(\bar{\mathcal{A}}^*)$ ▷ as described in Eqn. (12)
return L_{t+1}

$$\bar{\mathcal{A}}_S = \frac{1}{N\times T}\sum_{l=1}^{N}\sum_{k=0}^{T} Attn_{[l,k]}(\varepsilon,\varepsilon), \qquad \bar{\mathcal{A}}_X = \frac{1}{N\times T}\sum_{l=1}^{N}\sum_{k=0}^{T} Attn_{[l,k]}(\varepsilon,\tau),$$
$$\bar{\mathcal{A}}^* = \bar{\mathcal{A}}_S \circ \bar{\mathcal{A}}_X, \qquad \bar{\mathcal{A}}^* \in [0,1]^{H\times W}, \qquad \text{where } (H\times W) \text{ is the size of } \mathbf{I}_{t+1}. \tag{11}$$

Self-attention captures correlations among latent features, propagating the cross-attention to precise locations. Finally, as in Fig. 1e, different mappings produce desired prediction types:

$$L_{t+1} = \text{map}(\bar{\mathcal{A}}^*) = \begin{cases} \arg\max(\bar{\mathcal{A}}^*), & \text{if point} \\ \bar{\mathcal{A}}^* > 0, & \text{if segment} \\ (\min_i \beta, \min_j \beta, \max_i \beta, \max_j \beta), \quad \beta = \{(i,j) \mid \bar{\mathcal{A}}^*_{i,j} > 0\}, & \text{if box} \end{cases} \tag{12}$$

In summary, the entire diffusion-based tracking process involves the following steps. First, the indication of the object of interest at time t is injected as a condition by $p_\theta(\mathbf{z}_0^t|\tau)$, derived via Eqn. (3). Next, the video modeling process operates through the deterministic next-frame interpolation $p_\phi(\mathbf{z}_0^{t+1}|\mathbf{z}_0^t)$, as described in Subsection 4.3. Finally, the extraction of the object of interest in the next frame is performed via a so-called "reversed conditional process" $p_\theta^{-1}(\mathbf{z}_0^{t+1}|\tau)$, outlined in Alg. 3.

5 Experimental Results

5.1 Benchmarks and Metrics

TAP-Vid [18] formalizes the problem of long-term physical **Point Tracking**. It contains 31,951 points tracked on 1,219 real videos. Three evaluation metrics are *Occlusion Accuracy (OA)*, $< \delta^x_{avg}$ averaging position accuracy, and *Jaccard* @ δ quantifying occlusion and position accuracies.

PoseTrack21 [20] is similar to MOT17 [22]. In addition to estimating **Bounding Box** for each person, the body **Pose** needs to be estimated. Both keypoint-based and standard MOTA [106], IDF1 [107], and HOTA [108] evaluate the tracking performance for every keypoint visibility and subject identity.

DAVIS [24] and MOTS [26] are included to quantify the **Segmentation Tracking** performance. For the single-target dataset, evaluation metrics are Jaccard index $\mathcal{J}$, contour accuracy $\mathcal{F}$ and an overall $\mathcal{J}\&\mathcal{F}$ score [24]. For the multiple-target dataset, MOTSA and MOTSP [26] are equivalent to MOTA and MOTP, where the association metric measures the mask IoU instead of the bounding box IoU.

https://doi.org/10.52202/079017-2221

Finally, LaSOT [28] and GroOT [29] evaluate the **Referring Tracking** performance. The *Precision* and *Success* metrics are measured on LaSOT, while GroOT follows the evaluation protocol of MOT.

5.2 Implementation Details

We fine-tune the Latent Diffusion Models [13] inplace, follow [109, 110]. However, different from offline fixed batch retraining, our fine-tuning is performed online and auto-regressively between consecutive frames when a new frame is received. Our development builds on LDM [13] for settings with textual prompts and ADM [111] for localization settings, initialized by their publicly available pre-trained weights. The model is then fine-tuned using our proposed strategy for 500 steps with a learning rate of 3×10^{-5}. The model is trained on 4 NVIDIA Tesla A100 GPUs with a batch size of 1, comprising a pair of frames. We average the attention $\bar{A}_S$ and $\bar{A}_X$ in the interval $k \in [0, T \times 0.8]$ of the DDIM steps with the total timestep $T = 50$. For the first frame initialization, we employ YOLOX [112] as the detector, HRNet [113] as the pose estimator, and Mask2Former [114] as the segmentation model. We maintained a linear noise scheduler across all experiments, as it is the default in all available implementations and directly dependent on the number of diffusion steps, which is analyzed in the next subsection. Details for handling multiple objects are in Section D.

5.3 Ablation Study

Diffusion Steps. We systematically varied the number of diffusion steps (50, 100, 150, 200, 250) and analyzed their impact on performance and efficiency. Results show that we can reconstruct an image close to the origin with a timestep bound $T = 250$ in the reconstruction process of **DINTR**.

Table 3: The timestep bound T affects reconstruction quality.

T (steps)	50	100	150	200	250
MSE ↓	20.5	15.4	10.3	5.2	**0.04**
$\mathcal{J}\&\mathcal{F}$ ↑	75.4	75.8	76.0	76.3	**76.5**
Reconstruction time (s) ↓	6.2	12.7	17.5	23.6	28.7
Interpolation time (s) ↓	**3.2**	5.7	8.5	10.6	14.7

Alternative Approaches to the proposed **DINTR** modeling are discusses in this subsection. To substantiate the discussions, we include all ablation studies in Table 4, comparing against our base setting. These alternative settings are different interpolation operators as theoretically analyzed in Table 2, and different temporal modeling, including the Reconstruction process as visualized in Fig. 1c. Results demonstrate that our offset learning approach, which uses two anchor latents to deterministically guide the start and destination points, yields the best performance. This approach provides superior control over the interpolation process, resulting in more accurate and visually coherent output. For point tracking on TAP-Vid, **DINTR** achieves the highest scores, with AJ values ranging from 57.8 to 85.5 across different datasets. In pose tracking on PoseTrack, **DINTR** scores 82.5 mAP, significantly higher than other methods. For bounding box tracking on LaSOT, **DINTR** achieves the highest 0.74 precision and 0.70 success rate with text versus 0.60 precision and 0.58 success rate without text. In segment tracking on VOS, **DINTR** scores 75.7 for $\mathcal{J}\&\mathcal{F}$, 72.7 for $\mathcal{J}$, and 78.6 for $\mathcal{F}$, consistently outperforming other methods.

Table 4: Ablation studies of different temporal modeling alternatives (the second sub-block) and interpolation operators (the third sub-block) on point tracking (A), pose tracking (B), bounding box tracking with and without text (C), and segment tracking (D).

A. TAP-Vid	Kinetics		Kubric		DAVIS		RGB-Stacking	
	AJ	$< \delta^x_{avg}$	AJ	$< \delta^x_{avg}$	AJ	$< \delta^x_{avg}$	AJ	$< \delta^x_{avg}$
DINTR	**57.8**	**72.5**	**85.5**	**90.5**	**62.3**	**74.6**	**65.2**	**77.5**
(1c) Recon.	53.6	64.3	80.5	86.4	62.0	66.9	62.3	71.0
(2a) Linear	27.6	34.8	54.6	60.1	48.1	51.6	55.6	66.3
(2b) $\mathbf{z}_0^{t+1}$	34.1	43.3	64.9	63.9	51.6	54.8	59.7	60.3
(2c) $\mathbf{z}_0^t$	33.4	41.8	63.3	62.0	51.4	53.9	58.6	59.6

B. PoseTrack	**mAP**	**MOTA**	**IDF1**	**HOTA**
DINTR	**82.5**	**64.9**	**71.5**	**55.5**
(1c) Recon.	77.8	55.8	65.5	50.5
(2a) Linear	59.7	39.2	43.6	34.7
(2b) $\mathbf{z}_0^{t+1}$	69.1	43.6	55.1	40.7
(2c) $\mathbf{z}_0^t$	68.5	43.0	53.1	39.4

C. LaSOT	Precision	Success	Precision	Success
DINTR	**0.74**	**0.70**	**0.60**	**0.58**
(1c) Recon.	0.66	0.64	0.52	0.50
(2a) Linear	0.46	0.43	0.42	0.40
(2b) $\mathbf{z}_0^{t+1}$	0.52	0.49	0.46	0.45
(2c) $\mathbf{z}_0^t$	0.51	0.48	0.44	0.44

D. VOS	$\mathcal{J}\&\mathcal{F}$	$\mathcal{J}$	$\mathcal{F}$
DINTR	**75.7**	**72.7**	**78.6**
(1c) Recon.	73.9	71.8	76.1
(2a) Linear	43.8	46.1	41.5
(2b) $\mathbf{z}_0^{t+1}$	51.1	51.3	50.9
(2c) $\mathbf{z}_0^t$	50.5	51.0	49.9

https://doi.org/10.52202/079017-2221

Table 5: Point tracking performance against several methods on TAP-Vid [18].

TAP-Vid	Kinetics [115]			Kubric [116]			DAVIS [24]			RGB-Stacking [117]		
	AJ	$< \delta^x_{avg}$	OA	AJ	$< \delta^x_{avg}$	OA	AJ	$< \delta^x_{avg}$	OA	AJ	$< \delta^x_{avg}$	OA
COTR [118]	19.0	38.8	57.4	40.1	60.7	78.5	35.4	51.3	80.2	6.8	13.5	79.1
Kubric-VFS-Like [116]	40.5	59.0	80.0	51.9	69.8	84.6	33.1	48.5	79.4	57.9	72.6	91.9
RAFT [119]	34.5	52.5	79.7	41.2	58.2	86.4	30.0	46.3	79.6	44.0	58.6	90.4
PIPs [120]	35.1	54.8	77.1	59.1	74.8	88.6	42.0	59.4	82.1	37.3	51.0	91.6
TAP-Net [18]	46.6	60.9	85.0	65.4	77.7	93.0	38.4	53.1	82.3	59.9	72.8	90.4
TAPIR [30]	57.1	70.0	87.6	84.3	**91.8**	**95.8**	59.8	72.3	87.6	**66.2**	77.4	**93.3**
DINTR	**57.8**	**72.5**	**89.4**	**85.5**	90.5	95.2	**62.3**	**74.6**	**88.9**	65.2	**77.5**	91.6

The reconstruction-based method (1c) generally ranks second in performance across tasks. The decrease in performance for reconstruction is expected, as it does not transfer forward the final prediction to the next step. Instead, it reconstructs everything from raw noise at each step, as visualized in Fig. D.5. Although visual content can be well reconstructed, the lack of seamlessly transferred information between frames results in lower performance and reduced temporal coherence.

The performance difference between (2b) and (2c), which use a single anchor at either the starting latent point ($\mathbf{z}_0^t$) or destination latent point ($\mathbf{z}_0^{t+1}$) respectively, is minimal. However, we observed slightly higher effectiveness when controlling the destination point (2b) compared to the starting point (2b), suggesting that end-point guidance has a marginally stronger impact on overall interpolation quality. Linear blending (2a) consistently shows the lowest performance. Derivations of alternative operators blending (2a), learning from $\mathbf{z}_0^{t+1}$ (2b), learning from $\mathbf{z}_0^t$ (2c), and learning offset (2d) are theoretically proved to be equivalent as elaborated in Section C.

5.4 Comparisons to the State-of-the-Arts

Point Tracking. As presented in Table 5, our **DINTR** point model demonstrates competitive performance compared to prior works due to its thorough capture of local pixels and high-quality reconstruction of global context via the diffusion process. This results in the best performance on DAVIS and Kinetics datasets (88.9 and 89.4 OA). TAPIR [30] extracts features around the estimations rather than the global context. PIPs [120] and Tap-Net [18] lose flexibility by dividing the video into fixed segments. RAFT [119] cannot easily detect occlusions and makes accumulated errors due to per-frame tracking. COTR [118] struggles with moving objects as it operates on rigid scenes.

Pose Tracking. Table 6 compares our **DINTR** against other pose-tracking methods. Classic tracking methods, such as CorrTrack [121] and Tracktor++ [31], form appearance features with limited descriptiveness on keypoint representation. We also include DiffPose [122], another diffusion-based performer on the specific keypoint estimation task. The primary metric in this setting is the average precision computed for each joint and then averaged over all joints to obtain the final mAP. DiffPose [122] employs a similar diffusion-based generative process but operates on a different heatmap domain, achieving a similar performance on the pixel domain of our interpolation process.

Bounding Box Tracking. Table 7 shows the performance of single object tracking using bounding boxes or textual initialization. Similarly, Table 8 presents the performance of MOT using bounding boxes (left), against DiffussionTrack [3] and DiffMOT [4] or textual initialization (right), against MENDER [29] and MDETR+TrackFormer [129, 8]. Unlike DiffussionTrack [3] and DiffMOT [4], which are limited to specific initialization types, our approach allows flexible indicative injection from any type, improving unification capability, and achieving comparable performance. Moreover,

Table 6: Pose tracking performance against several methods on PoseTrack21 [20].

PoseTrack21	mAP	MOTA	IDF1	HOTA
CorrTrack [121]	72.3	63.0	66.5	51.1
Tracktor++ [31] w/ poses	71.4	63.3	69.3	52.2
CorrTrack [121] w/ ReID	72.7	63.8	66.5	52.7
Tracktor++ [31] w/ corr.	73.6	61.6	69.3	54.1
DCPose [123]	80.5	✗	✗	✗
FAMI-Pose [124]	81.2	✗	✗	✗
DiffPose [122]	**83.0**	✗	✗	✗
DINTR	82.5	**64.9**	**71.5**	**55.5**

Table 7: Single object tracking without (left) and with (right) textual prompt input.

LaSOT	Precision	Success	Precision	Success
SiamRPN++ [125]	0.50	0.45	✗	✗
GlobalTrack [126]	0.53	0.52	✗	✗
OCEAN [127]	0.57	0.56	✗	✗
UNICORN [10]	**0.74**	0.68	✗	✗
GTI [32]	✗	✗	0.47	0.47
AdaSwitcher [128]	✗	✗	0.55	0.51
DINTR	**0.74**	**0.70**	**0.60**	**0.58**

https://doi.org/10.52202/079017-2221

Table 8: Multiple object tracking without (left) and with (right) textual prompt input.

MOT17	MOTA	IDF1	HOTA	MT	ML	IDs
MOTR [9]	73.4	68.6	57.8	42.9%	19.1%	2439
TransMOT [130]	76.7	75.1	61.7	51.0%	16.4%	**2346**
UNICORN [10]	77.2	75.5	61.7	**58.7%**	**11.2%**	5379
DiffusionTrack [3]	77.9	73.8	60.8	–	–	–
DiffMOT [4]	**79.8**	**79.3**	**64.5**	–	–	–
DINTR	78.0	77.6	63.5	54.2%	14.6%	4878

GroOT	MOTA	IDF1	HOTA	AssA	DetA
MDETR+TFm	62.6	64.7	51.5	50.9	52.2
MENDER [29]	65.5	63.4	53.2	52.9	53.7
DINTR	**68.9**	**68.5**	**57.5**	**56.9**	**58.2**
(1c) *Reconstruct.*	63.0	58.6	48.4	48.0	49.1
(2b) $\mathbf{z}_0^{t+1}$	58.7	58.2	46.9	45.2	48.9

capturing global contexts via diffusion mechanics helps our model outperform MENDER and TrackFormer relying solely on spatial contexts formulated via transformer-based learnable queries.

Segment Tracking. Finally, Table 9 presents our segment tracking performance against *unified* methods [44, 10], *single-target* methods [43, 131], and *multiple-target* methods [37, 7, 8, 132]. Our **DINTR** achieves the best sMOTSA of 67.4, an accurate object tracking and segmentation. Unified methods perform the task separately, either using different branches [44] or stages [10]. It leads to a discrepancy in networks. Our **DINTR** that is both data- and process-unified avoids this shortcoming.

6 Conclusion

In conclusion, we have introduced a *Tracking-by-Diffusion* paradigm that reformulates the tracking framework based solely on visual iterative diffusion models. Unlike the existing denoising process, our **DINTR** offers a more seamless and faster approach to model temporal correspondences. This work has paved the way for efficient unified instance temporal modeling, especially object tracking.

Limitations. There is still a minor gap in performance to methods that incorporate *motion models*, *e.g.*, DiffMOT [4] with **2D** coordinate diffusion, as illustrated in Fig. 1b. However, our novel visual generative approach allows us to handle multiple representations in a unified manner rather than waste $5\times$ efforts on designing specialized models. As our approach introduces innovations from *feature representation* perspective, comparisons with advancements stemming from *heuristic optimizations*, such as ByteTrack [36], are not head-to-head as these are narrowly tailored increments for a specific type rather than paradigm shifts. However, exploring integrations between core representation and advancements offers promising performance. Specifically, final predictions are extracted by the so-called "reversed conditional process" $p_\theta^{-1}(\mathbf{z}_0^{t+1}|\tau)$ rather than sophisticated operations [133, 134]. Finally, time and resource consumption limit the practicality of Reconstruction. However, offline trackers continue to play a vital role in scenarios that demand comprehensive multimodality analysis.

Future Work & Broader Impacts. **DINTR** is a stepping stone towards more advanced and real-time visual *Tracking-by-Diffusion* in the future, especially to develop a new tracking approach that can manipulate visual contents [135] via the diffusion process or a foundation object tracking model. Specific future directions include formulating diffusion-based tracking approaches for open vocabulary [136], geometric constraints [11], camera motion [66, 137, 95], temporal displacement [5], object state [138], motion modeling [139, 6, 4], or new object representation [61] and management [140]. The proposed video modeling approach can be exploited for unauthorized surveillance and monitoring, or manipulating instance-based video content that could be used to spread misinformation.

Acknowledgment. This work is partly supported by NSF Data Science and Data Analytics that are Robust and Trusted (DART), USDA National Institute of Food and Agriculture (NIFA), and Arkansas Biosciences Institute (ABI) grants. We also acknowledge Trong-Thuan Nguyen for invaluable discussions and the Arkansas High-Performance Computing Center (AHPCC) for providing GPUs.

Table 9: Segment tracking performance on DAVIS [24] and MOTS [26].

VOS	$\mathcal{J}\&\mathcal{F}$	$\mathcal{J}$	$\mathcal{F}$
SiamMask [43]	56.4	54.3	58.5
Siam R-CNN [131]	70.6	66.1	75.0
UniTrack [44]	–	58.4	–
UNICORN [10]	69.2	65.2	73.2
DINTR	**75.4**	**72.5**	**78.4**

MOTS	sMOTSA	IDF1	MT	ML	IDSw
Track R-CNN [37]	40.6	42.4	38.7%	21.6%	567
TraDeS [7]	50.8	58.7	49.4%	18.3%	492
TrackFormer [8]	54.9	63.6	–	–	**278**
PointTrackV2 [132]	62.3	42.9	56.7%	12.5%	541
UNICORN [10]	65.3	65.9	64.9%	10.1%	398
DINTR	**67.4**	**66.4**	**66.5%**	**8.5%**	484

References

[1] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 1, 5

[2] Shoufa Chen, Peize Sun, Yibing Song, and Ping Luo. Diffusiondet: Diffusion model for object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19830–19843, 2023. 1, 2, 4

[3] Run Luo, Zikai Song, Lintao Ma, Jinlin Wei, Wei Yang, and Min Yang. Diffusiontrack: Diffusion model for multi-object tracking. *Proceedings of the AAAI Conference on Artificial Intelligence*, 2024. 1, 2, 3, 4, 9, 10

[4] Weiyi Lv, Yuhang Huang, Ning Zhang, Ruei-Sung Lin, Mei Han, and Dan Zeng. Diffmot: A real-time diffusion-based multiple object tracker with non-linear prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024. 1, 2, 3, 4, 9, 10

[5] Xingyi Zhou, Vladlen Koltun, and Philipp Krähenbühl. Tracking objects as points. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 474–490, 2020. 2, 3, 10

[6] Nicolai Wojke, Alex Bewley, and Dietrich Paulus. Simple online and realtime tracking with a deep association metric. In *2017 IEEE international conference on image processing (ICIP)*, pages 3645–3649. IEEE, 2017. 2, 10

[7] Jialian Wu, Jiale Cao, Liangchen Song, Yu Wang, Ming Yang, and Junsong Yuan. Track to detect and segment: An online multi-object tracker. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12352–12361, 2021. 2, 3, 4, 10

[8] Tim Meinhardt, Alexander Kirillov, Laura Leal-Taixe, and Christoph Feichtenhofer. Trackformer: Multi-object tracking with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8844–8854, 2022. 2, 3, 9, 10

[9] Fangao Zeng, Bin Dong, Yuang Zhang, Tiancai Wang, Xiangyu Zhang, and Yichen Wei. Motr: End-to-end multiple-object tracking with transformer. In *European Conference on Computer Vision*, pages 659–675. Springer, 2022. 2, 3, 10

[10] Bin Yan, Yi Jiang, Peize Sun, Dong Wang, Zehuan Yuan, Ping Luo, and Huchuan Lu. Towards grand unification of object tracking. In *European Conference on Computer Vision*, pages 733–751. Springer, 2022. 2, 3, 9, 10

[11] Patrick Dendorfer, Vladimir Yugay, Aljoša Ošep, and Laura Leal-Taixé. Quo vadis: Is trajectory forecasting the key towards long-term multi-object tracking? *Advances in Neural Information Processing Systems*, 35, 2022. 2, 4, 10

[12] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pages 2256–2265. PMLR, 2015. 2

[13] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 2, 4, 5, 8, 21

[14] Fitsum Reda, Janne Kontkanen, Eric Tabellion, Deqing Sun, Caroline Pantofaru, and Brian Curless. Film: Frame interpolation for large motion. In *European Conference on Computer Vision*, pages 250–266. Springer, 2022. 2

[15] Zhewei Huang, Tianyuan Zhang, Wen Heng, Boxin Shi, and Shuchang Zhou. Real-time intermediate flow estimation for video frame interpolation. In *European Conference on Computer Vision*, pages 624–642. Springer, 2022. 2

[16] Siddhant Jain, Daniel Watson, Eric Tabellion, Aleksander Hołyński, Ben Poole, and Janne Kontkanen. Video interpolation with diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024. 2

[17] Zhen Li, Zuo-Liang Zhu, Ling-Hao Han, Qibin Hou, Chun-Le Guo, and Ming-Ming Cheng. Amt: All-pairs multi-field transforms for efficient frame interpolation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9801–9810, 2023. 2

 https://doi.org/10.52202/079017-2221

[18] Carl Doersch, Ankush Gupta, Larisa Markeeva, Adrià Recasens, Lucas Smaira, Yusuf Aytar, João Carreira, Andrew Zisserman, and Yi Yang. Tap-vid: A benchmark for tracking any point in a video. *Advances in Neural Information Processing Systems*, 35:13610–13626, 2022. 3, 7, 9

[19] Mykhaylo Andriluka, Umar Iqbal, Eldar Insafutdinov, Leonid Pishchulin, Anton Milan, Juergen Gall, and Bernt Schiele. Posetrack: A benchmark for human pose estimation and tracking. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5167–5176, 2018. 3

[20] Andreas Doering, Di Chen, Shanshan Zhang, Bernt Schiele, and Juergen Gall. Posetrack21: A dataset for person search, multi-object tracking and multi-person pose tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20963–20972, 2022. 3, 7, 9

[21] L. Leal-Taixé, A. Milan, I. Reid, S. Roth, and K. Schindler. MOTChallenge 2015: Towards a benchmark for multi-target tracking. *arXiv:1504.01942 [cs]*, April 2015. arXiv: 1504.01942. 3

[22] A. Milan, L. Leal-Taixé, I. Reid, S. Roth, and K. Schindler. MOT16: A benchmark for multi-object tracking. *arXiv:1603.00831 [cs]*, March 2016. arXiv: 1603.00831. 3, 7

[23] Peize Sun, Jinkun Cao, Yi Jiang, Zehuan Yuan, Song Bai, Kris Kitani, and Ping Luo. Dancetrack: Multi-object tracking in uniform appearance and diverse motion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20993–21002, 2022. 3

[24] Federico Perazzi, Jordi Pont-Tuset, Brian McWilliams, Luc Van Gool, Markus Gross, and Alexander Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 724–732, 2016. 3, 7, 9, 10, 26, 27

[25] Linjie Yang, Yuchen Fan, and Ning Xu. Video instance segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5188–5197, 2019. 3

[26] Lorenzo Porzi, Markus Hofinger, Idoia Ruiz, Joan Serrat, Samuel Rota Bulo, and Peter Kontschieder. Learning multi-object tracking and segmentation from automatic annotations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6846–6855, 2020. 3, 7, 10

[27] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *2012 IEEE conference on computer vision and pattern recognition*, pages 3354–3361. IEEE, 2012. 3

[28] Heng Fan, Hexin Bai, Liting Lin, Fan Yang, Peng Chu, Ge Deng, Sijia Yu, Harshit, Mingzhen Huang, Juehuan Liu, et al. Lasot: A high-quality large-scale single object tracking benchmark. *International Journal of Computer Vision*, 129:439–461, 2021. 3, 8

[29] Pha Nguyen, Kha Gia Quach, Kris Kitani, and Khoa Luu. Type-to-track: Retrieve any object via prompt-based tracking. *Advances in Neural Information Processing Systems*, 36, 2023. 3, 5, 8, 9, 10

[30] Carl Doersch, Yi Yang, Mel Vecerik, Dilara Gokay, Ankush Gupta, Yusuf Aytar, Joao Carreira, and Andrew Zisserman. Tapir: Tracking any point with per-frame initialization and temporal refinement. *ICCV*, 2023. 3, 9

[31] Philipp Bergmann, Tim Meinhardt, and Laura Leal-Taixe. Tracking without bells and whistles. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 941–951, 2019. 2, 3, 9

[32] Zhengyuan Yang, Tushar Kumar, Tianlang Chen, Jingsong Su, and Jiebo Luo. Grounding-tracking-integration. *IEEE Transactions on Circuits and Systems for Video Technology*, 31(9):3433–3443, 2020. 3, 9

[33] Nicolai Wojke and Alex Bewley. Deep cosine metric learning for person re-identification. In *2018 IEEE winter conference on applications of computer vision (WACV)*, pages 748–756. IEEE, 2018. 3, 4

[34] Yongxin Wang, Kris Kitani, and Xinshuo Weng. Joint object detection and multi-object tracking with graph neural networks. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 13708–13715. IEEE, 2021. 2, 3

[35] Zhongdao Wang, Liang Zheng, Yixuan Liu, Yali Li, and Shengjin Wang. Towards real-time multi-object tracking. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 107–122. Springer, 2020. 3, 4

[36] Yifu Zhang, Peize Sun, Yi Jiang, Dongdong Yu, Fucheng Weng, Zehuan Yuan, Ping Luo, Wenyu Liu, and Xinggang Wang. Bytetrack: Multi-object tracking by associating every detection box. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022. 3, 10

[37] Paul Voigtlaender, Michael Krause, Aljosa Osep, Jonathon Luiten, Berin Balachandar Gnana Sekar, Andreas Geiger, and Bastian Leibe. Mots: Multi-object tracking and segmentation. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, pages 7942–7951, 2019. 3, 10

[38] Aljoša Ošep, Wolfgang Mehner, Paul Voigtlaender, and Bastian Leibe. Track, then decide: Category-agnostic vision-based multi-object tracking. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3494–3501. IEEE, 2018. 3

[39] Zhenbo Xu, Wei Zhang, Xiao Tan, Wei Yang, Huan Huang, Shilei Wen, Errui Ding, and Liusheng Huang. Segment as points for efficient online multi-object tracking and segmentation. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16*, pages 264–281. Springer, 2020. 3

[40] Yutao Cui, Tianhui Song, Gangshan Wu, and Limin Wang. Mixformerv2: Efficient fully transformer tracking. *Advances in Neural Information Processing Systems*, 36, 2024. 3

[41] Haojie Zhao, Xiao Wang, Dong Wang, Huchuan Lu, and Xiang Ruan. Transformer vision-language tracking via proxy token guided cross-modal fusion. *Pattern Recognition Letters*, 2023. 3

[42] Ruopeng Gao and Limin Wang. Memotr: Long-term memory-augmented transformer for multi-object tracking. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9901–9910, 2023. 3

[43] Qiang Wang, Li Zhang, Luca Bertinetto, Weiming Hu, and Philip HS Torr. Fast online object tracking and segmentation: A unifying approach. In *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition*, pages 1328–1338, 2019. 3, 10

[44] Zhongdao Wang, Hengshuang Zhao, Ya-Li Li, Shengjin Wang, Philip Torr, and Luca Bertinetto. Do different tracking tasks require different appearance models? *Advances in Neural Information Processing Systems*, 34:726–738, 2021. 3, 4, 10

[45] Christoph Feichtenhofer, Axel Pinz, and Andrew Zisserman. Detect to track and track to detect. In *Proceedings of the IEEE international conference on computer vision*, pages 3038–3046, 2017. 2

[46] Qiankun Liu, Qi Chu, Bin Liu, and Nenghai Yu. Gsm: Graph similarity model for multi-object tracking. In *IJCAI*, pages 530–536, 2020. 2

[47] Guillem Brasó and Laura Leal-Taixé. Learning a neural solver for multiple object tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6247–6257, 2020. 2

[48] Jerome Berclaz, Francois Fleuret, Engin Turetken, and Pascal Fua. Multiple object tracking using k-shortest paths optimization. *IEEE transactions on pattern analysis and machine intelligence*, 33(9):1806–1819, 2011. 2

[49] Kha Gia Quach, Pha Nguyen, Huu Le, Thanh-Dat Truong, Chi Nhan Duong, Minh-Triet Tran, and Khoa Luu. Dyglip: A dynamic graph model with link prediction for accurate multi-camera multiple object tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13784–13793, 2021. 2

[50] Andrea Hornakova, Roberto Henschel, Bodo Rosenhahn, and Paul Swoboda. Lifted disjoint paths with application in multiple object tracking. In *International Conference on Machine Learning*, pages 4364–4375. PMLR, 2020. 2

[51] Roberto Henschel, Laura Leal-Taixé, Daniel Cremers, and Bodo Rosenhahn. Improvements to frank-wolfe optimization for multi-detector multi-object tracking. *arXiv preprint arXiv:1705.08314*, 8, 2017. 2

[52] Siyu Tang, Mykhaylo Andriluka, Bjoern Andres, and Bernt Schiele. Multiple people tracking by lifted multicut and person re-identification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3539–3548, 2017. 2

[53] Duy MH Nguyen, Roberto Henschel, Bodo Rosenhahn, Daniel Sonntag, and Paul Swoboda. Lmgp: Lifted multicut meets geometry projections for multi-camera multi-object tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8866–8875, 2022. 2

 https://doi.org/10.52202/079017-2221

[54] Qian Yu, Gérard Medioni, and Isaac Cohen. Multiple target tracking using spatio-temporal markov chain monte carlo data association. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2007. 2

[55] Margret Keuper, Siyu Tang, Bjoern Andres, Thomas Brox, and Bernt Schiele. Motion segmentation & multiple object tracking by correlation co-clustering. *IEEE transactions on pattern analysis and machine intelligence*, 42(1):140–153, 2018. 2

[56] Chanho Kim, Fuxin Li, Arridhana Ciptadi, and James M Rehg. Multiple hypothesis tracking revisited. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 4696–4704, 2015. 2

[57] Hao Sheng, Yang Zhang, Jiahui Chen, Zhang Xiong, and Jun Zhang. Heterogeneous association graph fusion for target association in multiple object tracking. *IEEE Transactions on Circuits and Systems for Video Technology*, 29(11):3269–3280, 2018. 2

[58] Hao Jiang, Sidney Fels, and James J Little. A linear programming approach for multiple object tracking. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2007. 2

[59] Hamed Pirsiavash, Deva Ramanan, and Charless C Fowlkes. Globally-optimal greedy algorithms for tracking a variable number of objects. In *CVPR 2011*, pages 1201–1208. IEEE, 2011. 2

[60] Li Zhang, Yuan Li, and Ramakant Nevatia. Global data association for multi-object tracking using network flows. In *2008 IEEE conference on computer vision and pattern recognition*, pages 1–8. IEEE, 2008. 2

[61] Jathushan Rajasegaran, Georgios Pavlakos, Angjoo Kanazawa, and Jitendra Malik. Tracking people by predicting 3d appearance, location and pose. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2740–2749, 2022. 2, 10

[62] Peng Chu and Haibin Ling. Famnet: Joint learning of feature, affinity and multi-dimensional assignment for online multiple object tracking. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6172–6181, 2019. 2

[63] Jiangmiao Pang, Linlu Qiu, Xia Li, Haofeng Chen, Qi Li, Trevor Darrell, and Fisher Yu. Quasi-dense similarity learning for multiple object tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 164–173, 2021. 2

[64] Ergys Ristani and Carlo Tomasi. Features for multi-target multi-camera tracking and re-identification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6036–6046, 2018. 2

[65] Laura Leal-Taixé, Cristian Canton-Ferrer, and Konrad Schindler. Learning by tracking: Siamese cnn for robust target association. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 33–40, 2016. 2

[66] Nir Aharon, Roy Orfaig, and Ben-Zion Bobrovsky. Bot-sort: Robust associations multi-pedestrian tracking. *arXiv preprint arXiv:2206.14651*, 2022. 2, 10

[67] Jinkun Cao, Jiangmiao Pang, Xinshuo Weng, Rawal Khirodkar, and Kris Kitani. Observation-centric sort: Rethinking sort for robust multi-object tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9686–9696, 2023. 2

[68] Laura Leal-Taixé, Gerard Pons-Moll, and Bodo Rosenhahn. Everybody needs somebody: Modeling social and grouping behavior on a linear programming multiple people tracker. In *2011 IEEE international conference on computer vision workshops (ICCV workshops)*, pages 120–127. IEEE, 2011. 2

[69] Stefano Pellegrini, Andreas Ess, Konrad Schindler, and Luc Van Gool. You'll never walk alone: Modeling social behavior for multi-target tracking. In *2009 IEEE 12th international conference on computer vision*, pages 261–268. IEEE, 2009. 2

[70] Paul Scovanner and Marshall F Tappen. Learning pedestrian dynamics from the real world. In *2009 IEEE 12th International Conference on Computer Vision*, pages 381–388. IEEE, 2009. 2

[71] Kota Yamaguchi, Alexander C Berg, Luis E Ortiz, and Tamara L Berg. Who are you with and where are you going? In *CVPR 2011*, pages 1345–1352. IEEE, 2011. 2

[72] Anton Andriyenko and Konrad Schindler. Multi-target tracking by continuous energy minimization. In *CVPR 2011*, pages 1265–1272. IEEE, 2011. 2

[73] Long Chen, Haizhou Ai, Zijie Zhuang, and Chong Shang. Real-time multiple people tracking with deeply learned candidate selection and person re-identification. In *2018 IEEE international conference on multimedia and expo (ICME)*, pages 1–6. IEEE, 2018. 2

[74] Alexandre Alahi, Kratarth Goel, Vignesh Ramanathan, Alexandre Robicquet, Li Fei-Fei, and Silvio Savarese. Social lstm: Human trajectory prediction in crowded spaces. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 961–971, 2016. 2

[75] Alexandre Robicquet, Amir Sadeghian, Alexandre Alahi, and Silvio Savarese. Learning social etiquette: Human trajectory prediction in crowded scenes. In *European Conference on Computer Vision (ECCV)*, volume 2, page 5, 2016. 2

[76] Laura Leal-Taixé, Michele Fenzi, Alina Kuznetsova, Bodo Rosenhahn, and Silvio Savarese. Learning an image-based motion context for multiple people tracking. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3542–3549, 2014. 2

[77] Pavel Tokmakov, Jie Li, Wolfram Burgard, and Adrien Gaidon. Learning to track with object permanence. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10860–10869, 2021. 2

[78] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 3, 4

[79] Yuang Zhang, Tiancai Wang, and Xiangyu Zhang. Motrv2: Bootstrapping end-to-end multi-object tracking by pretrained object detectors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22056–22065, 2023. 3

[80] Jiarui Cai, Mingze Xu, Wei Li, Yuanjun Xiong, Wei Xia, Zhuowen Tu, and Stefano Soatto. Memot: Multi-object tracking with memory. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8090–8100, 2022. 3

[81] Xinyu Zhou, Pinxue Guo, Lingyi Hong, Jinglun Li, Wei Zhang, Weifeng Ge, and Wenqiang Zhang. Reading relevant feature from global representation memory for visual object tracking. *Advances in Neural Information Processing Systems*, 36, 2024. 3

[82] Eric Hedlin, Gopal Sharma, Shweta Mahajan, Hossam Isack, Abhishek Kar, Andrea Tagliasacchi, and Kwang Moo Yi. Unsupervised semantic correspondence using stable diffusion. *Advances in Neural Information Processing Systems*, 36, 2023. 3

[83] Konpat Preechakul, Nattanat Chatthee, Suttisak Wizadwongsa, and Supasorn Suwajanakorn. Diffusion autoencoders: Toward a meaningful and decodable representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10619–10629, 2022. 3

[84] Junyi Zhang, Charles Herrmann, Junhwa Hur, Luisa Polania Cabrera, Varun Jampani, Deqing Sun, and Ming-Hsuan Yang. A tale of two features: Stable diffusion complements dino for zero-shot semantic correspondence. *Advances in Neural Information Processing Systems*, 36, 2023. 3

[85] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel HAZIZA, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024. 3

[86] Grace Luo, Lisa Dunlap, Dong Huk Park, Aleksander Holynski, and Trevor Darrell. Diffusion hyperfeatures: Searching through time and space for semantic correspondence. In *Advances in Neural Information Processing Systems*, volume 36, 2023. 3, 4

[87] Luming Tang, Menglin Jia, Qianqian Wang, Cheng Perng Phoo, and Bharath Hariharan. Emergent correspondence from image diffusion. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. 3, 4

[88] Mingi Kwon, Jaeseok Jeong, and Youngjung Uh. Diffusion models already have a semantic latent space. In *The Eleventh International Conference on Learning Representations*, 2023. 3

[89] Sarthak Mittal, Korbinian Abstreiter, Stefan Bauer, Bernhard Schölkopf, and Arash Mehrjou. Diffusion based representation learning. In *International Conference on Machine Learning*, pages 24963–24982. PMLR, 2023. 4

 https://doi.org/10.52202/079017-2221

[90] Charan D Prakash and Lina J Karam. It gan do better: Gan-based detection of objects on images with varying quality. *IEEE Transactions on Image Processing*, 30:9220–9230, 2021. 4

[91] Pengxiang Li, Zhili Liu, Kai Chen, Lanqing Hong, Yunzhi Zhuge, Dit-Yan Yeung, Huchuan Lu, and Xu Jia. Trackdiffusion: Multi-object tracking data generation via diffusion models. *arXiv preprint arXiv:2312.00651*, 2023. 4

[92] Quang Nguyen, Truong Vu, Anh Tran, and Khoi Nguyen. Dataset diffusion: Diffusion-based synthetic data generation for pixel-level semantic segmentation. *Advances in Neural Information Processing Systems*, 36, 2024. 4

[93] Agrim Gupta, Justin Johnson, Li Fei-Fei, Silvio Savarese, and Alexandre Alahi. Social gan: Socially acceptable trajectories with generative adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2255–2264, 2018. 4

[94] Koichi Namekata, Amirmojtaba Sabour, Sanja Fidler, and Seung Wook Kim. Emerdiff: Emerging pixel-level semantic knowledge in diffusion models. In *The Twelfth International Conference on Learning Representations*, 2024. 4

[95] Pha Nguyen, Kha Gia Quach, Chi Nhan Duong, Son Lam Phung, Ngan Le, and Khoa Luu. Multi-camera multi-object tracking on the move via single-stage global association approach. *Pattern Recognition*, page 110457, 2024. 4, 10

[96] Pha Nguyen, Kha Gia Quach, John Gauch, Samee U Khan, Bhiksha Raj, and Khoa Luu. Utopia: Unconstrained tracking objects without preliminary examination via cross-domain adaptation. *arXiv preprint arXiv:2306.09613*, 2023. 4

[97] Thanh-Dat Truong, Chi Nhan Duong, Ashley Dowling, Son Lam Phung, Jackson Cothren, and Khoa Luu. Crovia: Seeing drone scenes from car perspective via cross-view adaptation. *arXiv preprint arXiv:2304.07199*, 2023. 4

[98] Nikita Karaev, Ignacio Rocco, Benjamin Graham, Natalia Neverova, Andrea Vedaldi, and Christian Rupprecht. Cotracker: It is better to track together. *arXiv preprint arXiv:2307.07635*, 2023. 4

[99] Guillaume Le Moing, Jean Ponce, and Cordelia Schmid. Dense optical tracking: Connecting the dots. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024. 4

[100] Yuxi Xiao, Qianqian Wang, Shangzhan Zhang, Nan Xue, Sida Peng, Yujun Shen, and Xiaowei Zhou. Spatialtracker: Tracking any 2d pixels in 3d space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024. 4

[101] Tobias Fischer, Thomas E Huang, Jiangmiao Pang, Linlu Qiu, Haofeng Chen, Trevor Darrell, and Fisher Yu. Qdtrack: Quasi-dense similarity learning for appearance-only multiple object tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023. 4

[102] Dongming Wu, Wencheng Han, Tiancai Wang, Xingping Dong, Xiangyu Zhang, and Jianbing Shen. Referring multi-object tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14633–14642, 2023. 5

[103] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021. 5

[104] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pages 234–241. Springer, 2015. 5, 21

[105] Eric Heitz, Laurent Belcour, and Thomas Chambon. Iterative α-(de) blending: A minimalist deterministic diffusion model. In *ACM SIGGRAPH 2023 Conference Proceedings*, pages 1–8, 2023. 7

[106] Keni Bernardin and Rainer Stiefelhagen. Evaluating multiple object tracking performance: the clear mot metrics. *EURASIP Journal on Image and Video Processing*, 2008:1–10, 2008. 7

[107] Ergys Ristani, Francesco Solera, Roger Zou, Rita Cucchiara, and Carlo Tomasi. Performance measures and a data set for multi-target, multi-camera tracking. In *European conference on computer vision*, pages 17–35. Springer, 2016. 7

[108] Jonathon Luiten, Aljosa Osep, Patrick Dendorfer, Philip Torr, Andreas Geiger, Laura Leal-Taixé, and Bastian Leibe. Hota: A higher order metric for evaluating multi-object tracking. *International journal of computer vision*, 129:548–578, 2021. 7

[109] Jay Zhangjie Wu, Yixiao Ge, Xintao Wang, Stan Weixian Lei, Yuchao Gu, Yufei Shi, Wynne Hsu, Ying Shan, Xiaohu Qie, and Mike Zheng Shou. Tune-a-video: One-shot tuning of image diffusion models for text-to-video generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7623–7633, 2023. 8, 25

[110] Chenyang Qi, Xiaodong Cun, Yong Zhang, Chenyang Lei, Xintao Wang, Ying Shan, and Qifeng Chen. Fatezero: Fusing attentions for zero-shot text-based video editing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023. 8, 25

[111] Prafulla Dhariwal and Alex Nichol. Diffusion models beat gans on image synthesis. *Neural Information Processing Systems*, 2021. 8

[112] Zheng Ge, Songtao Liu, Feng Wang, Zeming Li, and Jian Sun. Yolox: Exceeding yolo series in 2021. *arXiv preprint arXiv:2107.08430*, 2021. 8

[113] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5693–5703, 2019. 8

[114] Bowen Cheng, Alex Schwing, and Alexander Kirillov. Per-pixel classification is not all you need for semantic segmentation. *Advances in Neural Information Processing Systems*, 34:17864–17875, 2021. 8

[115] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308, 2017. 9

[116] Klaus Greff, Francois Belletti, Lucas Beyer, Carl Doersch, Yilun Du, Daniel Duckworth, David J Fleet, Dan Gnanapragasam, Florian Golemo, Charles Herrmann, et al. Kubric: A scalable dataset generator. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3749–3761, 2022. 9

[117] Alex X Lee, Coline Manon Devin, Yuxiang Zhou, Thomas Lampe, Konstantinos Bousmalis, Jost Tobias Springenberg, Arunkumar Byravan, Abbas Abdolmaleki, Nimrod Gileadi, David Khosid, et al. Beyond pick-and-place: Tackling robotic stacking of diverse shapes. In *Conference on Robot Learning*, pages 1089–1131. PMLR, 2022. 9

[118] Wei Jiang, Eduard Trulls, Jan Hosang, Andrea Tagliasacchi, and Kwang Moo Yi. Cotr: Correspondence transformer for matching across images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6207–6217, 2021. 9

[119] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*, pages 402–419. Springer, 2020. 9

[120] Adam W Harley, Zhaoyuan Fang, and Katerina Fragkiadaki. Particle video revisited: Tracking through occlusions using point trajectories. In *European Conference on Computer Vision*, pages 59–75. Springer, 2022. 9

[121] Umer Rafi, Andreas Doering, Bastian Leibe, and Juergen Gall. Self-supervised keypoint correspondences for multi-person pose estimation and tracking in videos. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XX 16*, pages 36–52. Springer, 2020. 9

[122] Runyang Feng, Yixing Gao, Tze Ho Elden Tse, Xueqing Ma, and Hyung Jin Chang. Diffpose: Spatiotemporal diffusion model for video-based human pose estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14861–14872, 2023. 9

[123] Zhenguang Liu, Runyang Feng, Haoming Chen, Shuang Wu, Yixing Gao, Yunjun Gao, and Xiang Wang. Temporal feature alignment and mutual information maximization for video-based human pose estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11006–11016, 2022. 9

[124] Zhenguang Liu, Haoming Chen, Runyang Feng, Shuang Wu, Shouling Ji, Bailin Yang, and Xun Wang. Deep dual consecutive network for human pose estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 525–534, 2021. 9

[125] Bo Li, Junjie Yan, Wei Wu, Zheng Zhu, and Xiaolin Hu. High performance visual tracking with siamese region proposal network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8971–8980, 2018. 9

 https://doi.org/10.52202/079017-2221

[126] Lianghua Huang, Xin Zhao, and Kaiqi Huang. Globaltrack: A simple and strong baseline for long-term tracking. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 11037–11044, 2020. 9

[127] Zhipeng Zhang, Houwen Peng, Jianlong Fu, Bing Li, and Weiming Hu. Ocean: Object-aware anchor-free tracking. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXI 16*, pages 771–787. Springer, 2020. 9

[128] Xiao Wang, Xiujun Shu, Zhipeng Zhang, Bo Jiang, Yaowei Wang, Yonghong Tian, and Feng Wu. Towards more flexible and accurate object tracking with natural language: Algorithms and benchmark. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13763–13773, 2021. 9

[129] Aishwarya Kamath, Mannat Singh, Yann LeCun, Gabriel Synnaeve, Ishan Misra, and Nicolas Carion. Mdetr-modulated detection for end-to-end multi-modal understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1780–1790, 2021. 9

[130] Peng Chu, Jiang Wang, Quanzeng You, Haibin Ling, and Zicheng Liu. Transmot: Spatial-temporal graph transformer for multiple object tracking. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 4870–4880, 2023. 10

[131] Paul Voigtlaender, Jonathon Luiten, Philip HS Torr, and Bastian Leibe. Siam r-cnn: Visual tracking by re-detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6578–6588, 2020. 10

[132] Zhenbo Xu, Wei Yang, Wei Zhang, Xiao Tan, Huan Huang, and Liusheng Huang. Segment as points for efficient and effective online multi-object tracking and segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(10):6424–6437, 2021. 10

[133] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020. 10

[134] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017. 10

[135] Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, Devi Parikh, Sonal Gupta, and Yaniv Taigman. Make-a-video: Text-to-video generation without text-video data. In *The Eleventh International Conference on Learning Representations*, 2023. 10

[136] Siyuan Li, Tobias Fischer, Lei Ke, Henghui Ding, Martin Danelljan, and Fisher Yu. Ovtrack: Open-vocabulary multiple object tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5567–5577, 2023. 10

[137] Pha Nguyen, Kha Gia Quach, Chi Nhan Duong, Ngan Le, Xuan-Bac Nguyen, and Khoa Luu. Multi-camera multiple 3d object tracking on the move for autonomous vehicles. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 2569–2578, June 2022. 10

[138] ShiJie Sun, Naveed Akhtar, XiangYu Song, HuanSheng Song, Ajmal Mian, and Mubarak Shah. Simultaneous detection and tracking with motion modelling for multiple object tracking. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 626–643, 2020. 10

[139] Alex Bewley, Zongyuan Ge, Lionel Ott, Fabio Ramos, and Ben Upcroft. Simple online and realtime tracking. In *2016 IEEE International Conference on Image Processing (ICIP)*, pages 3464–3468. IEEE, 2016. 10

[140] Daniel Stadler and Jurgen Beyerer. Improving multiple pedestrian tracking by track management and occlusion handling. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10958–10967, 2021. 10

[141] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-or. Prompt-to-prompt image editing with cross-attention control. In *The Eleventh International Conference on Learning Representations*, 2023. 21

[142] Huanran Chen, Yinpeng Dong, Shitong Shao, Zhongkai Hao, Xiao Yang, Hang Su, and Jun Zhu. Your diffusion model is secretly a certifiably robust classifier. *arXiv preprint arXiv:2402.02316*, 2024. 21

[143] Alexander C Li, Mihir Prabhudesai, Shivam Duggal, Ellis Brown, and Deepak Pathak. Your diffusion model is secretly a zero-shot classifier. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2206–2217, 2023. 21

 https://doi.org/10.52202/079017-2221

Appendices

A Glossary

Table A.10: Notations used throughout the paper.

$\mathbf{I}_t$	Current processing frame (image), $\mathbf{I}_t \in \mathbb{R}^{H \times W \times 3}$
$\mathbf{I}_{t+1}$	Next frame (image) in the processing video
L_t	Indicator representation in the current processing frame $\mathbf{I}_t$ (*e.g.* point, bounding box, segment, or *text*)
L_{t+1}	Location in the current processing frame $\mathbf{I}_t$ (*e.g.* point, bounding box, or segment)
$\mathcal{E}(\mathbf{I})$	Visual encoder $\mathcal{E}$ extracting visual features
$\mathcal{E}(\mathbf{I}_t)[L_t]$	Pooled visual features of the current frame at the indicated location
$\mathcal{D}(\mathbf{z}_0)$	Visual decoder decoding latent feature to image
θ	Network parameters
ϵ	A noise variable, $\epsilon \sim \mathcal{N}(0, 1)$
τ	Indicator representation
$\varepsilon_\theta(\mathbf{z}_k)$	Denoising autoencoders, *i.e.*, U-Net blocks
$\phi_\theta(\mathbf{z}_k)$	Interpolation network, having the same structure as ε_θ
$\|\cdot\|_2^2$	L^2 norm
$\mathbf{z}_0, \ldots, \mathbf{z}_k, \ldots, \mathbf{z}_T$	Latent variables of the noise sampling process
$\widehat{\mathbf{z}}_0, \ldots, \widehat{\mathbf{z}}_k, \ldots, \widehat{\mathbf{z}}_T$	Latent variables of the reconstructive interpolation process
α_k	The scheduling parameter
$\mathcal{Q}(\cdot)$	Noise sampling process
$\mathcal{P}_{\varepsilon_\theta}(\cdot)$	Reconstruction/Denoising process, configured by ε_θ
$\mathcal{P}_{\phi_\theta}(\cdot)$	Interpolation process, configured by ϕ_θ
$\mathcal{T}_\theta(\cdot)$	Indication feature extractor
$\mathbb{E}_{\varepsilon_\theta} L(\cdot)$	Expectation of a loss function $L(\cdot)$ with respect to ϵ_θ
$D_{KL}(P\|Q)$	Kullback-Leibler divergence of P and Q
$q(\mathbf{z}_k^t \| \mathbf{z}_{k-1}^t)$	Conditional probability of $\mathbf{z}_k^t$ given $\mathbf{z}_{k-1}^t$
$p_\varepsilon(\mathbf{z}_{k-1}^t \| \mathbf{z}_k^t)$	Conditional probability of denoising $\mathbf{z}_{k-1}^t$ given $\mathbf{z}_k^t$, configured by ε
$p_\phi(\widehat{\mathbf{z}}_{k-1}^t \| \widehat{\mathbf{z}}_k^t)$	Conditional probability of interpolating $\widehat{\mathbf{z}}_{k-1}^t$ given $\widehat{\mathbf{z}}_k^t$, configured by ϕ
$\left(\mathcal{P}_{\phi_\theta}(\cdot) \to \mathcal{P}_{\phi_\theta}(\cdot)\right)$	Induction process
$\bar{\mathcal{A}}_S$	Average self-attention maps among visual features in U-Net
$\bar{\mathcal{A}}_X$	Average cross-attention maps among visual features in U-Net
$\bar{\mathcal{A}}^*$	Element-wise product of self- and cross-attention

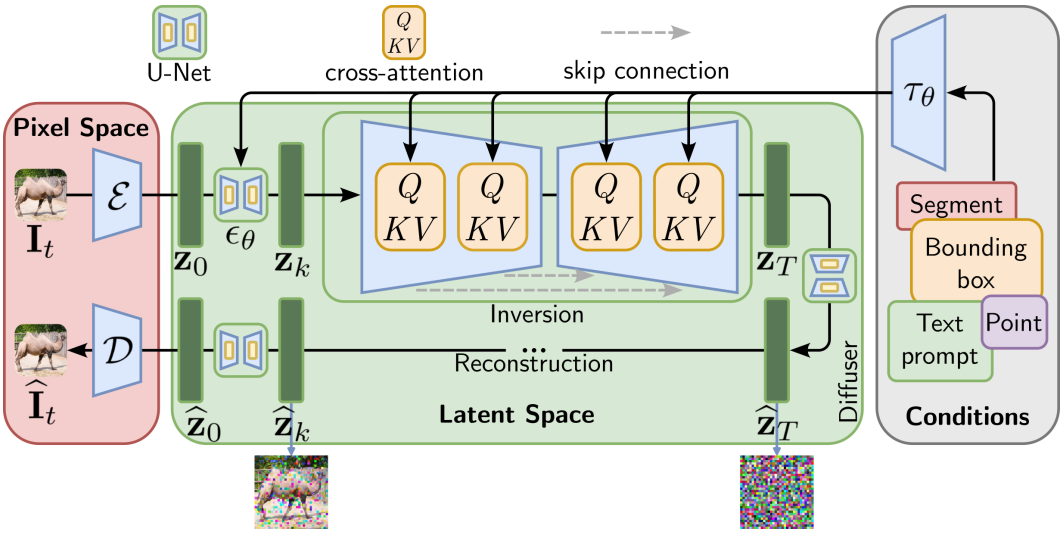


Figure B.3: The conditional LDMs utilizes U-Net [104] blocks. First, a clean image $\mathbf{I}_k$ is converted to a noisy latent $\mathbf{z}_k$ via the noise sampling process $\mathcal{Q}(\cdot)$ (top branch). Then, well-structured regions are reconstructed from that extremely noisy input via the denoising/reconstruction process $\mathcal{P}_{\varepsilon_\theta}(\cdot)$ (bottom branch). Additionally, conditions can be added as indicators of the regions of interest. While the figure style is adapted from LDMs [13], we made a distinct change reflecting the *injected* sampling process, following Prompt-to-Prompt [141].

B Overall Framework

Salient Representation. The ability of the diffuser to, first, *convert a clean image to a noisy latent*, having no recognizable pattern from its origin, and then, *reconstruct well-structured regions from extremely noisy input*, indicates that the diffuser produces powerful semantic contexts [142, 143].

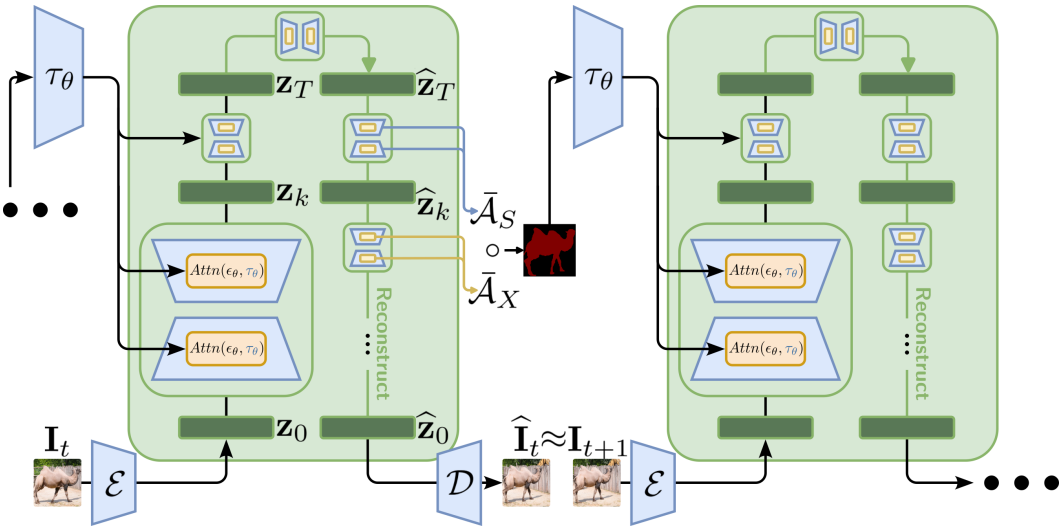


Figure B.4: Our proposed autoregressive framework constructed via the diffusion mechanics for temporal modeling. The current frame is input to the encoder $\mathcal{E}(\mathbf{I}_t)$ to produce an initial latent $\mathbf{z}_0$. The sampling process $\mathcal{Q}(\cdot)$ adds noises into the latent in a sequence of T steps. Next, reconstruction process $\mathcal{P}_{\varepsilon_\theta}(\cdot)$ is manipulated through KL divergence optimization *w.r.t.* $\mathbf{z}_{k-1}^{t+1}$. This shapes the reconstructed image $\widehat{\mathbf{I}}_t$ to be more similar to the future frame $\mathbf{I}_{t+1}$. Finally, the location of the targets can be extracted by spatial correspondences, exhibited by the attention maps $\bar{\mathcal{A}}_S$ and $\bar{\mathcal{A}}_X$.

https://doi.org/10.52202/079017-2221

In other words, the diffuser can embed semantic alignments, producing coherent predictions between two templates. To leverage this capability, we first consider the generated image $\widehat{\mathbf{I}}_t$ in the diffusion process. Identifying correspondences on the pixel domain can be achieved if:

$$dist\Big(\mathcal{E}(\mathbf{I}_t), \mathcal{E}(\widehat{\mathbf{I}}_t)\Big) = 0 \text{ is } \textit{optimal} \text{ from Eqn. (2), then } dist\Big(\mathcal{E}(\mathbf{I}_t)[L_t], \mathcal{E}(\widehat{\mathbf{I}}_t)[L_t]\Big) = 0. \quad \text{(B.13)}$$

We extract the latent features $\mathbf{z}_k$ of their intermediate U-Net blocks at a specific time step k during both processes. This is then utilized to establish injected correspondences between the input image $\mathbf{I}_k$ and the generated image $\widehat{\mathbf{I}}_k$.

Injected Condition. By incorporating conditional indicators into the Inversion process, we can guide the model to focus on a particular object of interest. This conditional input, represented as points, poses (*i.e.*, structured set of points), segments, bounding boxes, or even textual prompts, acts as an indicator to *inject the region of interest into the clean latent*, which we want the model to recognize in the reconstructed latent.

These two remarks support the visual diffusion process in capturing and semantically manipulating features for representing and distinguishing objects, as illustrated in Fig. B.3. Additionally, Fig. B.4 presents the autoregressive process that injects and extracts internal states to identify the target regions holding the correspondence temporally.

C Derivations of Equivalent Interpolative Operators

This section derives the variant formulations introduced in Subsection 4.3.

C.1 Interpolated Samples

In the field of image processing, an interpolated data point is defined as a weighted combination of known data points through a blending operation controlled by a weighted parameter α_k:

$$\widehat{\mathbf{z}}_k^{t+1} = \alpha_k\, \mathbf{z}_0^t + (1-\alpha_k)\, \mathbf{z}_0^{t+1}. \quad \text{(C.14)}$$

We can thus rewrite its known samples $\mathbf{z}_0^{t+1}$ and $\mathbf{z}_0^t$ in the following way:

$$\mathbf{z}_0^{t+1} = \frac{\widehat{\mathbf{z}}_k^{t+1}}{1-\alpha_k} - \frac{\alpha_k\, \mathbf{z}_0^t}{1-\alpha_k}, \quad \text{(C.15)}$$

$$\mathbf{z}_0^{t} = \frac{\widehat{\mathbf{z}}_k^{t+1}}{\alpha_k} - \frac{(1-\alpha_k)\, \mathbf{z}_0^{t+1}}{\alpha_k}. \quad \text{(C.16)}$$

C.2 Linear Blending (2a)

In the vanilla version of the algorithm, a blended sample of parameter α_k is obtained by blending $\mathbf{z}_0^{t+1}$ and $\mathbf{z}_0^t$, as similar as Eqn. (C.14):

$$\widehat{\mathbf{z}}_{k-1}^{t+1} = \alpha_{k-1}\, \mathbf{z}_0^t + (1-\alpha_{k-1})\, \mathbf{z}_0^{t+1}. \quad \text{(C.17)}$$

To train our interpolation approach using this operator, because the accumulativeness property does not hold, then the step-wise loss as defined in Eqn. (5) has to be employed. As a result, this is equivalent to the reconstruction approach *Reconstruct.* described in Eqn. (4) and reported in Subsection 5.3.

C.3 Learning from $\mathbf{z}_0^{t+1}$ (2b)

By expanding $\mathbf{z}_0^t$ from Eqn. (C.17) using Eqn. (C.16), we obtain:

$$\begin{aligned}
\widehat{\mathbf{z}}_{k-1}^{t+1} &= (1-\alpha_{k-1})\,\mathbf{z}_0^{t+1} + \alpha_{k-1}\,\mathbf{z}_0^t, \\
&= (1-\alpha_{k-1})\,\mathbf{z}_0^{t+1} + \alpha_{k-1}\left(\frac{\widehat{\mathbf{z}}_k^{t+1}}{\alpha_k} - \frac{(1-\alpha_k)\,\mathbf{z}_0^{t+1}}{\alpha_k}\right), \\
&= \left(1-\alpha_{k-1} - \frac{\alpha_{k-1}\,(1-\alpha_k)}{\alpha_k}\right)\mathbf{z}_0^{t+1} + \frac{\alpha_{k-1}}{\alpha_k}\,\widehat{\mathbf{z}}_k^{t+1}, \\
&= \left(\frac{\alpha_k - \alpha_k\,\alpha_{k-1} - \alpha_{k-1}\,(1-\alpha_k)}{\alpha_k}\right)\mathbf{z}_0^{t+1} + \frac{\alpha_{k-1}}{\alpha_k}\,\widehat{\mathbf{z}}_k^{t+1}, \\
&= \left(1-\frac{\alpha_{k-1}}{\alpha_k}\right)\mathbf{z}_0^{t+1} + \frac{\alpha_{k-1}}{\alpha_k}\,\widehat{\mathbf{z}}_k^{t+1}, \\
&= \mathbf{z}_0^{t+1} + \frac{\alpha_{k-1}}{\alpha_k}\left(\widehat{\mathbf{z}}_k^{t+1} - \mathbf{z}_0^{t+1}\right).
\end{aligned} \tag{C.18}$$

Inductive Process. With the base case $\widehat{\mathbf{z}}_T^{t+1} = \mathbf{z}_0^t$, the transition is accumulative within the inductive data interpolation:

$$\begin{aligned}
&k \in \{T-1, \ldots, 1\}, \\
&\Big(\underbrace{\mathcal{P}_{\phi_\theta}\big(\mathbf{z}_0^{t+1} + \frac{\alpha_k}{\alpha_{k+1}}(\widehat{\mathbf{z}}_{k+1}^{t+1} - \mathbf{z}_0^{t+1}), k, \tau\big)}_{\widehat{\mathbf{z}}_k^{t+1}} \rightarrow \mathcal{P}_{\phi_\theta}\big(\mathbf{z}_0^{t+1} + \frac{\alpha_{k-1}}{\alpha_k}(\widehat{\mathbf{z}}_k^{t+1} - \mathbf{z}_0^{t+1}), k-1, \tau\big)\Big).
\end{aligned} \tag{C.19}$$

C.4 Learning from $\mathbf{z}_0^t$ (2c)

By expanding $\mathbf{z}_0^{t+1}$ from Eqn. (C.17) using Eqn. (C.15), we obtain:

$$\begin{aligned}
\widehat{\mathbf{z}}_{k-1}^{t+1} &= (1-\alpha_{k-1})\,\mathbf{z}_0^{t+1} + \alpha_{k-1}\,\mathbf{z}_0^t, \\
&= (1-\alpha_{k-1})\left(\frac{\widehat{\mathbf{z}}_k^{t+1}}{1-\alpha_k} - \frac{\alpha_k\,\mathbf{z}_0^t}{1-\alpha_k}\right) + \alpha_{k-1}\,\mathbf{z}_0^t, \\
&= \left(\alpha_{k-1} - \frac{(1-\alpha_{k-1})\,\alpha_k}{1-\alpha_k}\right)\mathbf{z}_0^t + \frac{1-\alpha_{k-1}}{1-\alpha_k}\,\widehat{\mathbf{z}}_k^{t+1}, \\
&= \left(\frac{\alpha_{k-1}\,(1-\alpha_k) - (1-\alpha_{k-1})\,\alpha_k}{1-\alpha_k}\right)\mathbf{z}_0^t + \frac{1-\alpha_{k-1}}{1-\alpha_k}\,\widehat{\mathbf{z}}_k^{t+1}, \\
&= \left(\frac{1-\alpha_k - (1-\alpha_{k-1})}{1-\alpha_k}\right)\mathbf{z}_0^t + \frac{1-\alpha_{k-1}}{1-\alpha_k}\,\widehat{\mathbf{z}}_k^{t+1}, \\
&= \left(1-\frac{1-\alpha_{k-1}}{1-\alpha_k}\right)\mathbf{z}_0^t + \frac{1-\alpha_{k-1}}{1-\alpha_k}\,\widehat{\mathbf{z}}_k^{t+1}, \\
&= \mathbf{z}_0^t + \frac{1-\alpha_{k-1}}{1-\alpha_k}\left(\widehat{\mathbf{z}}_k^{t+1} - \mathbf{z}_0^t\right).
\end{aligned} \tag{C.20}$$

Inductive Process. With the base case $\widehat{\mathbf{z}}_T^{t+1} = \mathbf{z}_0^t$, the transition is accumulative within the inductive data interpolation:

$$\begin{aligned}
&k \in \{T-1, \ldots, 1\}, \\
&\Big(\underbrace{\mathcal{P}_{\phi_\theta}\big(\mathbf{z}_0^t + \frac{1-\alpha_k}{1-\alpha_{k+1}}(\widehat{\mathbf{z}}_{k+1}^{t+1} - \mathbf{z}_0^t), k, \tau\big)}_{\widehat{\mathbf{z}}_k^{t+1}} \rightarrow \mathcal{P}_{\phi_\theta}\big(\mathbf{z}_0^t + \frac{1-\alpha_{k-1}}{1-\alpha_k}(\widehat{\mathbf{z}}_k^{t+1} - \mathbf{z}_0^t), k-1, \tau\big)\Big).
\end{aligned} \tag{C.21}$$

https://doi.org/10.52202/079017-2221

Due to the absence of the deterministic property and the target term $\mathbf{z}_0^{t+1}$, the loss in Eqn. (7) becomes the sole objective guiding the learning process toward the target. Consequently, we prefer to perform the interpolation operator (2b) in Subsection 5.3, which is theoretically equivalent to this operator.

C.5 Learning Offset (2d)

By rewriting $\alpha_{k-1} = \alpha_{k-1} + \alpha_k - \alpha_k$ in the definition of $\widehat{\mathbf{z}}_{k-1}^{t+1}$, we obtain:

$$\begin{aligned}\widehat{\mathbf{z}}_{k-1}^{t+1} &= (1-\alpha_{k-1})\,\mathbf{z}_0^{t+1} + \alpha_{k-1}\,\mathbf{z}_0^{t},\\ &= (1-\alpha_{k-1}+\alpha_k-\alpha_k)\,\mathbf{z}_0^{t+1} + (\alpha_{k-1}+\alpha_k-\alpha_k)\;\mathbf{z}_0^{t},\\ &= (1-\alpha_k)\,\mathbf{z}_0^{t+1} + \alpha_k\,\mathbf{z}_0^{t} + (\alpha_{k-1}-\alpha_k)\;\left(\mathbf{z}_0^{t}-\mathbf{z}_0^{t+1}\right).\end{aligned} \tag{C.22}$$

Replace $(1-\alpha_k)\,\mathbf{z}_0^{t+1} + \alpha_k\,\mathbf{z}_0^{t}$ by $\widehat{\mathbf{z}}_k^{t+1}$ from Eqn. (C.14), we obtain:

$$\begin{aligned}\widehat{\mathbf{z}}_{k-1}^{t+1} &= \widehat{\mathbf{z}}_k^{t+1} + (\alpha_{k-1}-\alpha_k)\;\left(\mathbf{z}_0^{t}-\mathbf{z}_0^{t+1}\right),\\ &= \widehat{\mathbf{z}}_k^{t+1} + (\alpha_k-\alpha_{k-1})\;\left(\mathbf{z}_0^{t+1}-\mathbf{z}_0^{t}\right),\\ &= \widehat{\mathbf{z}}_k^{t+1} + \frac{k-(k-1)}{T}\;\left(\mathbf{z}_0^{t+1}-\mathbf{z}_0^{t}\right).\end{aligned} \tag{C.23}$$

By multiplying the step $\left(\mathbf{z}_0^{t+1}-\mathbf{z}_0^{t}\right)$ by a larger factor (*e.g.*, T), the scaled step maintain their magnitude and not to become too small when propagated through many layers. Then we obtain:

$$\widehat{\mathbf{z}}_{k-1}^{t+1} \propto \widehat{\mathbf{z}}_k^{t+1} + \left(\mathbf{z}_0^{t+1}-\mathbf{z}_0^{t}\right), \quad \text{signified} \tag{C.24}$$

$$\propto \widehat{\mathbf{z}}_k^{t+1} + \left(\mathbf{z}_{k-1}^{t+1}-\mathbf{z}_k^{t}\right), \tag{C.25}$$

$$= \widehat{\mathbf{z}}_k^{t+1} + \Big(\mathcal{Q}\left(\mathbf{z}_0^{t+1}, k-1\right) - \mathcal{Q}\left(\mathbf{z}_0^{t}, k\right)\Big), \quad \text{as in L4 of Alg. 2.} \tag{C.26}$$

Inductive Process. With the base case $\widehat{\mathbf{z}}_T^{t+1} = \mathbf{z}_0^{t}$, the transition is accumulative within the inductive data interpolation:

$$\begin{aligned}&k \in \{T-1,\ldots,1\},\\ &\Big(\underbrace{\mathcal{P}_{\phi_\theta}\left(\widehat{\mathbf{z}}_{k+1}^{t+1} + (\mathbf{z}_k^{t+1}-\mathbf{z}_{k+1}^{t}), k, \tau\right)}_{\widehat{\mathbf{z}}_k^{t+1}} \to \mathcal{P}_{\phi_\theta}\left(\widehat{\mathbf{z}}_k^{t+1} + (\mathbf{z}_{k-1}^{t+1}-\mathbf{z}_k^{t}), k-1, \tau\right)\Big).\end{aligned} \tag{C.27}$$

D Technical Details

Multiple-Target Handling. Our method processes multiple object tracking by first concatenating all target representations into a joint input tensor during both the Inversion and Reconstruction passes through the diffusion model. Specifically, given M targets, indexed by i, each with a indicator representation L_t^i, we form the concatenated input:

$$\mathcal{T} = \Big[\mathcal{T}_\theta(L_t^0) \| \ldots \| \mathcal{T}_\theta(L_t^i) \| \ldots \| \mathcal{T}_\theta(L_t^{M-1})\Big]. \tag{D.28}$$

where $[\,\cdot \,\|\, \cdot\,]$ is the concatenation operation.

This allows encoding interactions and contexts across all targets simultaneously while passing through the same encoder, decoder modules, and processes. After processing the concatenated output $\mathcal{P}_{\phi_\theta}(\mathbf{z}_0^t, T, \mathcal{T})$, we split it back into the individual target attention outputs using their original index order:

$$\bar{\mathcal{A}}_X = \Big[\bar{\mathcal{A}}_X^0 \| \ldots \| \bar{\mathcal{A}}_X^i \| \ldots \| \bar{\mathcal{A}}_X^{M-1}\Big], \quad \bar{\mathcal{A}}_X \in [0,1]^{M\times H\times W}. \tag{D.29}$$

So each $\bar{\mathcal{A}}_X^i$ contains the refined cross-attention for target i after joint diffusion with the full set of targets. This approach allows the model to enable target-specific decoding. The indices linking inputs

to corresponding outputs are crucial for maintaining identity and predictions during the sequence of processing steps.

Textual Prompt Handling. This setting differs from the other four indicator types, where L_0 comes from a dedicated object detector. Instead, we leverage the unique capability of diffusion models to generate from text prompts [109, 110]. Specifically, we initialize L_0 using a textual description as the conditioning input. From this textual L_0, our process generates an initial set of bounding box proposals as L_1. These box proposals then propagate through the subsequent iterative processes to refine into the next $L_2, \ldots, L_{|\mathbf{V}|-2}$ tracking outputs.

Pseudo-code for One-shot Training. Alg. D.4 and Alg. D.5 are the pseudo-code for our fine-tuning and operating algorithms in the proposed approach within the *Tracking-by-Diffusion* paradigm, respectively. The pseudo-code provides an overview of the steps involved in our inplace fine-tuning.

Algorithm D.4 The one-shot fine-tuning pipeline of Reconstruction process

Input: $\mathbf{I}_t, \mathbf{I}_{t+1}, \mathcal{T} \leftarrow [\tau_\theta(L_t^0) \| \ldots \| \tau_\theta(L_t^{M-1})], T \leftarrow 50$
$\mathbf{z}_0 \leftarrow \mathcal{E}(\mathbf{I}_t)$
$\mathbf{x}_0 \leftarrow \mathcal{E}(\mathbf{I}_{t+1})$
$\mathbf{z}_T \leftarrow \mathcal{Q}(\mathbf{z}_0, T)$ % *injected Inversion*
$L_{\text{ELBO}} \leftarrow \text{KL}\big(\mathcal{Q}(\mathbf{x}_{T-1}, T) \| \mathcal{P}(\mathbf{z}_T, T, \mathcal{T})\big)$ % ℓ_T
for $k \in \{T, \ldots, 2\}$ **do**
  L_{ELBO} += $\text{KL}\big(\mathcal{Q}(\mathbf{x}_{k-2}, k) \| \mathcal{P}(\widehat{\mathbf{z}}_k, k, \mathcal{T})\big)$ % ℓ_{k-1}
end for
L_{ELBO} −= $\log \mathcal{P}(\widehat{\mathbf{z}}_1)$ % ℓ_0
Take gradient descent step on L_{ELBO}

Algorithm D.5 The tracker operation

Input: Video $\mathbf{V}$, set of tracklets $\mathbf{T} \leftarrow \{L_0^0, \ldots, L_0^{M-1}\}, \beta = 4, T \leftarrow 50$
for $t \in \{0, \ldots, |\mathbf{V}| - 2\}$ **do**
  Draw $(\mathbf{I}_t, \mathbf{I}_{t+1}) \in \mathbf{V}$
  $\mathcal{T} \leftarrow [\tau_\theta(L_t^0) \| \ldots \| \tau_\theta(L_t^{M-1})]$ % $\mathcal{T}$ *not change if* L_t^i *is textual prompt*
  $finetuning(\mathbf{I}_t, \mathbf{I}_{t+1}, \mathcal{T})$ % *via* Alg. D.4
  $\widehat{\mathbf{z}}_T \leftarrow \mathcal{P}(\mathbf{z}_T, T, \mathcal{T})$
  for $k \in \{T, \ldots, 1\}$ **do**
    if $k \in [1, T \times 0.8]$ **then**
      $\mathcal{A}_S$ += $\sum_{l=1}^{N} Attn_{l,k}(\epsilon_\theta, \epsilon_\theta)$
      $\mathcal{A}_X$ += $\sum_{l=1}^{N} Attn_{l,k}(\epsilon_\theta, \tau_\theta)$
    end if
    $\widehat{\mathbf{z}}_k \leftarrow \mathcal{P}(\widehat{\mathbf{z}}_{k+1}, k, \mathcal{T})$
  end for
  $\bar{\mathcal{A}}_S \leftarrow \frac{1}{N \times T} \sum_{k=1}^{T} \mathcal{A}_S$
  $\bar{\mathcal{A}}_X \leftarrow \frac{1}{N \times T} \sum_{k=1}^{T} \mathcal{A}_X$
  $\bar{\mathcal{A}}^* \leftarrow (\bar{\mathcal{A}}_S)^\beta \circ \bar{\mathcal{A}}_X$
  $[L_{t+1}^0 \| \ldots \| L_{t+1}^{M-1}] \leftarrow mapping(\bar{\mathcal{A}}^*)$ % *via* Eqn. (12)
  $\mathbf{T} \leftarrow \{L_{t+1}^0, \ldots, L_{t+1}^{M-1}\}$
end for

Process Visualization. Fig. D.5 and Fig. D.6 are visualizing the two proposed diffusion-based processes that are utilized in our tracker framework.

https://doi.org/10.52202/079017-2221

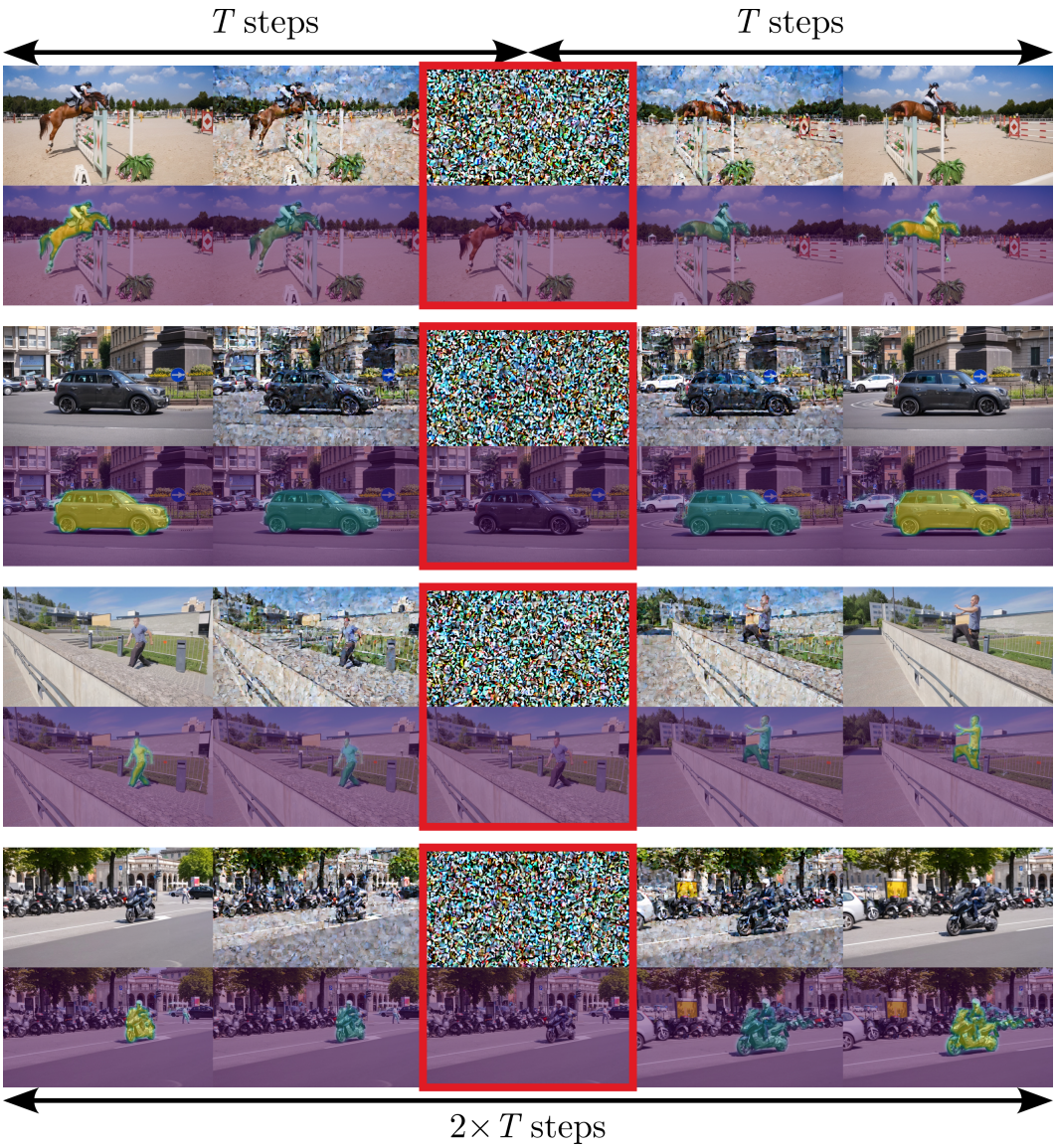


Figure D.5: The visualization depicts the diffusion-based Reconstruction process on the DAVIS benchmark [24]. Unlike the interpolation process in Fig. D.6, where internal states are efficiently transferred between frames, the reconstruction process samples visual contents from extreme noise (middle column), and attention maps cannot be transferred. Although visual content can be reconstructed, **the lack of seamlessly transferred information** between frames results in lower performance and reduced temporal coherence as in Tables 5, 6, 7, 8, and 9.

https://doi.org/10.52202/079017-2221

Figure D.6: Visualization of the diffusion-based Interpolation process on the DAVIS benchmark [24]. Different from the reconstruction process in Fig. D.5, where each frame is processed independently, visual contents (top), internal states, and attention maps (bottom) are efficiently transferred from the previous frame to the next frame. This **seamless transfer of information** between frames results in more consistent and stable tracking, as the model can leverage temporal coherence.

https://doi.org/10.52202/079017-2221

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: See **Contributions** in Section 1. Assumptions in diffusion models are clearly stated, including the conditional mechanism, and diffusion mechanics (*i.e.*, the denoising process).

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Please see **Limitations** in Section 6.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [NA]

Justification: This paper does not include *pure* theoretical results, but the equivalences or derivations of formulas are included.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Please find the Subsection 5.2.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

https://doi.org/10.52202/079017-2221

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: The techniques presented in this work are the intellectual property of [Affiliation], and the organization intends to seek patent coverage for the disclosed process.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: Please see Subsection 4.3 and Subsection 5.2.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)

- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Please see Subsection 5.2.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Please see **Future Work & Broader Impacts** in Section 6.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

https://doi.org/10.52202/079017-2221

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The original papers that produced the code package or dataset are cited.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

https://doi.org/10.52202/079017-2221