Autoregressive Temporal Modeling for Advanced *Tracking-by-Diffusion*

Pha Nguyen[1*], Rishi Madhok[2], Bhiksha Raj[3], Khoa Luu[1*]

[1]Department of EECS, University of Arkansas, Fayetteville, AR, USA.
[2]Microsoft, Redmond, WA, USA.
[3]Carnegie Mellon University, Pittsburgh, PA, USA.

*Corresponding author(s). E-mail(s): panguyen@uark.edu; khoaluu@uark.edu;
Contributing authors: rishi.madhok@microsoft.com; bhiksha@cs.cmu.edu;

Abstract

Object tracking is a widely studied computer vision task with video and instance analysis applications. While paradigms such as *tracking-by-regression*, *-detection*, *-attention* have advanced the field, generative modeling offers new potential. Although some studies explore the generative process in instance-based understanding tasks, they rely on prediction refinement in the coordinate space rather than the visual domain. Instead, this paper presents *Tracking-by-Diffusion*, a novel paradigm for object tracking in video, leveraging visual generative models via the perspective of autoregressive models. This paradigm demonstrates broad applicability across point, box, and mask modalities while uniquely enabling textual guidance. We present DIFTracker, a framework that utilizes iterative latent variable diffusion models to redefine tracking as a next-frame reconstruction task. Our approach uniquely combines spatial and temporal dependencies in video data, offering a unified solution that encompasses existing tracking paradigms within a single Inversion-Reconstruction process. DIFTracker operates online and autoregressively, enabling flexible instance-based video understanding. It allows us to overcome difficulties in variable-length video understanding encountered by video-inflated models and perform superior performance on seven benchmarks across five modalities. This paper not only introduces a new perspective on visual autoregressive modeling in understanding sequential visual data, specifically videos, but also provides robust theoretical validations and demonstrates broader applications in visual tracking and computer vision.

Keywords: Autoregressive Models, Diffusion Models, Visual Tracking, Unification

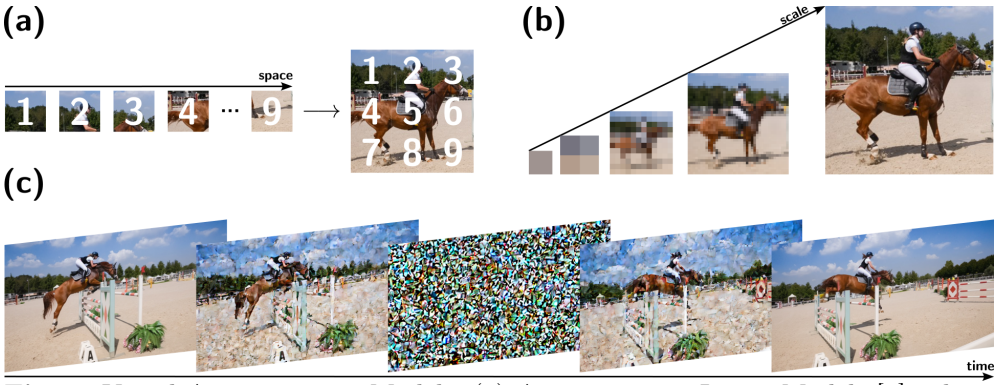

Fig. 1: Visual Autoregressive Models. (a) Autoregressive Image Models [1], where input images are split into non-overlapping patches and embedded linearly. (b) Visual Autoregressive Models via Next-scale Prediction [2], where multi-scale token maps are autoregressively generated from coarse to fine scales. (c) Our concept of Autoregressive Models for Temporal Modeling powered by Diffusion Models [3, 4].

1 Introduction

Autoregressive models have emerged as a cornerstone in machine learning, offering a simple yet potent approach to predicting the next values based on their predecessors. The power of autoregressive models lies in their ability to capture complex dependencies within sequential data, making them invaluable tools for tasks that require understanding and generating structured information. As a result, this technique has found recent advanced applications, including image generation and natural language processing. When applied to images, this method facilitates the creation of visual content through the sequential generation of visual tokens in a raster-scan order (*i.e.*, progressing from left to right and top to bottom [1], as in Fig. 1a), or from coarse to fine scales (*i.e.* lower to higher resolutions [2], as in Fig. 1b). Extending this concept to sequential visual data, such as video, presents exciting opportunities and unique challenges. Video can be conceptualized as a temporal sequence of images, where each frame depends not only on its spatial context but also on the temporal context provided by preceding frames. Therefore, visual autoregressive models applied to temporal modeling must account for spatial and temporal dependencies, as illustrated in Fig. 1c.

In this work, we explore the mechanics of visual generative models, specifically diffusion models, and the visual autoregressive paradigm in a sequential visual data context, particularly video for instance-based understanding. We specifically focus on the practical task of object tracking, a long-standing challenge in computer vision with widespread applications in video analysis and instance-level understanding. Numerous tracking paradigms have been explored over past decades, including *tracking-by-regression* [5], *-detection* [6], *-segmentation* [7] and two more recently *tracking-by-attention* [8, 9], *-unification* [10] paradigms. Recently, generative modeling has achieved great success in this field, and it offers several promising new perspectives, including denoising sampling bounding boxes to final prediction [11, 12], or sampling future trajectories [13]. While

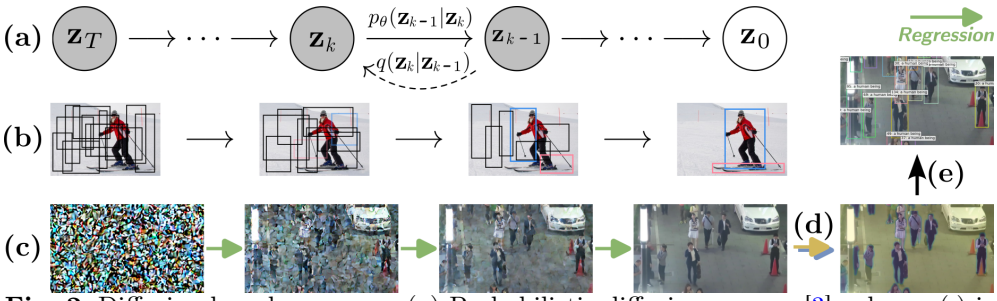

Fig. 2: Diffusion-based processes. (a) Probabilistic diffusion process [3], where $q(\cdot)$ is Inversion and $p_\theta(\cdot)$ is Reconstruction. (b) Diffusion process in 2D coordinate space [11, 12]. (c) Our purely visual diffusion-based process, learned to *reconstruct* ***real frames***, named *Regression* $reg(\cdot)$, formulated in our proposed DIFTracker (Eqn. (11)). (d) and (e) are heatmap extraction (Eqn. (13)) and mapping prediction (Eqn. (14)), respectively.

these studies explore the generative process in instance-based understanding tasks, they rely on prediction refinement in the coordinate space (latent variables $\mathbf{z}_k \in \mathbb{R}^2$ for coordinates). These coordinate-based approaches fail to leverage the conditioning mechanism [14] of Latent Diffusion Models, which are principally capable of modeling unified conditional distributions. They can not encode and decode more complex prediction types, such as hierarchically structured points, dense segmentation masks, or abstract text outputs, as illustrated in Fig. 2b and validated in Table 1.

To redefine the visual autoregressive paradigm on video data for this task, we firstly present a novel tracking framework purely based on *visual* iterative latent variable diffusion models [41, 14], defining the novel and true *Tracking-by-Diffusion* paradigm by formulating latent variables $\mathbf{z}_k \in \mathbb{R}^{H\times W\times 3}$ for video frames. This paradigm demonstrates broad applications across tracking modalities, including points, bounding boxes, and segmentation masks. Uniquely, the generative reconstruction process enables conditional guidance from textual prompts [42, 43] during tracking. By injecting conditional inputs, we demonstrate that our proposed framework unifies capabilities spanning all existing tracking paradigms within a single Inversion-Reconstruction process. Our paradigm reformulates the classic object tracking task to the novel task of *next-frame reconstruction*, our key novelty, and leverages the visual understanding ability of diffusion models. To the best of our knowledge, *no existing method has formulated* ***tracking via autoregressive reconstruction*** as our proposed approach in this work.

Moreover, our proposed approach in this *Tracking-by-Diffusion* paradigm, DIFTracker, models the temporal domain by injecting the Inversion process and manipulating the Reconstruction process. Finally, by formulating the diffusion-based temporal modeling to operate *online* and *auto-regressively* (*i.e.* next-frame reconstruction), DIFTracker enables flexibility for instance-based video understanding, specifically tracking. In addition, this efficient integration surmounts limitations faced by prior inflated diffusion modeling [44, 42] as will be discussed in §2.3.

Contributions. Overall, this paper formulates and presents a novel *Tracking-by-Diffusion* paradigm from the novel perspective of *visual autoregressive generative models*.

Table 1: Comparison of supported modalities, paradigms, and mechanisms of SOTA tracking methods. **Indicator Modalities** defines the representation to indicate tracking targets with their corresponding datasets: **TAP-Vid** [15], **PoseTrack** [16, 17], **MOT** [18], **VOS** [19], **VIS** [20], **MOTS** [21] **KITTI** [22], **SOT** [23], **GroOT** [24]. Blue text indicates **single-target**, while green text indicates **multi-target benchmarks**.

Method	Paradigm	Mechanism*	Indicator Modalities				
			Point	Pose	Box	Segment	Text
TAPIR [25]	Regression	*Iter.* Refinement	**TAP**	✗	✗	✗	✗
Tracktor++ [26]		Regression Head	✗	✗	**MOT**	✗	✗
CenterTrack [5]		Offset Prediction	✗	✗	**MOT**	✗	✗
GTI [27]		*Rgn-Tpl Integ.*	✗	✗	**SOT**	✗	**LaSOT**
DeepSORT [28]	Detection	Cascade *Assoc.*	✗	✗	**MOT**	✗	✗
GSDT [29]		Relation Graph	✗	✗	**MOT**	✗	✗
JDE [30]		Multi-Task	✗	✗	**MOT**	✗	✗
ByteTrack [31]		Two-stage *Assoc.*	✗	✗	**MOT**	✗	✗
TrackR-CNN [32]	Segmentation	3D Convolution	✗	✗	✗	**MOTS**	✗
MOTSNet [21]		Mask-Pooling	✗	✗	✗	**MOTS**	✗
CAMOT [33]		Hypothesis Select	✗	✗	✗	**KITTI**	✗
PointTrack [34]		*Seg.* as Points	✗	✗	✗	**MOTS**	✗
MixFormerV2 [35]	Attention	Mixed Attention	✗	✗	**SOT**	✗	✗
MeMOTR [36]		Memory *Aug.*	✗	✗	**MOT**	✗	✗
TrackFormer [8]		Set Prediction	✗	✗	**MOT**	✗	✗
TransVLT [37]		X-Modal Fusion	✗	✗	**SOT**	✗	**LaSOT**
MENDER [24]		Tensor *Decomp.*	✗	✗	**MOT**	✗	**GroOT**
SiamMask [38]	Unification	Variant Head	✗	✗	**SOT**	**VOS**	✗
TraDeS [7]		Cost Volume	✗	✗	**MOT**	**VIS** **MOTS**	✗
UNICORN [10]		Unified *Embed.*	✗	✗	**SOT** **MOT**	**VOS** **MOTS**	✗
UniTrack [39]		Primitive Level	✗	**PoseTrack**	**SOT** **MOT**	**VOS** **MOTS**	✗
DiffusionTrack [12]	**Diffusion**	Denoised *Coord.*	✗	✗	**MOT**	✗	✗
DiffMOT [40]		*Motion* Predictor	✗	✗	**MOT**	✗	✗
DIFTracker		Manipulated *Rec.*	**TAP**	**PoseTrack**	**SOT** **MOT**	**VOS** **MOTS**	**SOT** **GroOT**

* *Iter.*: Iterative. *Rgn-Tpl Integ.*: Region-Template Integration. *Assoc.*: Association. *Seg.*: Segmentation. *Aug.*: Augmentation. X: Cross. *Decomp.*: Decomposition. *Embed.*: Embedding. *Coord.*: **2D** Coordinate. *Motion*: **2D** Motion ($\mathbf{z}_k \in \mathbb{R}^2$). *Rec.*: *Visual* Reconstruction ($\mathbf{z}_k \in \mathbb{R}^{H \times W \times 3}$).

Such a paradigm is coped with robust theoretical validations and demonstrates wider applications and greater flexibility compared to the existing trackers. Additionally, this paradigm features no explicit object location training but focuses primarily on frame reconstruction. Our proposed DIFTracker approach in this paradigm performs State-of-the-Art (SOTA) performance and outperforms inflated video diffusion models [44, 43] across *seven tracking benchmarks* comprising *five modalities*. The following sections further discuss its capabilities, formulation, and empirical evaluations.

2 Related Work

2.1 Tracking Paradigms

In this section, we categorize the tracking methods according to their respective paradigms, including *tracking-by-regression*, *-detection* or *association*, *-segmentation*, *-attention* and *-unification*, and summarize these recent studies in Table 1.

Tracking-by-Regression refines future object positions based on visual features without linking detections across frames. Previous methods [26, 45] rely on the regression branch of object features in specific regions. Additionally, CenterTrack [5] represents objects via center points and a distance-based matching algorithm. Still, it lacks explicit object identity and global tracking, necessitating the integration of appearance re-identification methods [26], motion models [46], as well as both traditional and advanced graph techniques [47].

Tracking-by-Detection forms object trajectories by linking detections over frames, treating the task as an optimization problem.

Graph-based methods formulate the tracking problem as a maximum flow or minimum cost optimization [48]. These methods utilize a variety of techniques, such as distance-based [49, 50, 51], association graphs [52], learned models [53], motion information [54], general-purpose solvers [55], multi-cuts [56, 57], weighted graph labeling [58], edge lifting [59], trainable graph neural networks [47, 29], and link prediction [60]. Despite their versatility, these approaches were hampered by the high cost of optimization, which limits their applications in online tracking.

Additionally, *Appearance*-based methods leverage robust image recognition frameworks to track objects. These techniques depend on similarity measures derived from twin neural networks [61], learned re-identification features [62, 63], detection candidate selection [64], affinity estimation [64], or 3D appearance and pose [65]. However, these appearance-based models face significant challenges in highly crowded and occluded environments.

On the other hand, *Motion* modeling is leveraged for location estimation [66, 67, 68] directly from trajectory sequences, based on constant velocity assumptions [69, 70], the social force model [67, 71, 72, 73], trajectory forecasting [13], observation-centric manner [74], or camera motion [75]. Recently, motion models learned from data [76] effectively track associations across frames. These models still need help to project complex 3D motions [77] into 2D space.

Tracking-by-Segmentation leverages detailed pixel information and addresses the challenges of crowded scenes and unclear backgrounds. Methods include mask-based [33] with 3D convolutions [32], mask pooling layers [21], point cloud representations [34], and cost volumes [7]. However, its reliance on segmented multiple object tracking data often necessitates bounding boxes.

Tracking-by-Attention applies the attention mechanism [78] to link detections with tracks at the feature level, being represented as tokens. TrackFormer [8] approaches tracking as a unified prediction task using attention, especially 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 buffer [80] and memory augmentation [36]. Recently, MENDER [24] represents another stride, a transformer architecture with tensor decomposition to facilitate object tracking through descriptions.

Tracking-by-Unification. Recently, some new attempts have been proposed to design unified frameworks supporting multiple tracking tasks. Specifically, SiamMask [38] pioneered joint single object tracking (SOT) and video object segmentation (VOS). Similarly, TraDeS [7] solves multiple object tracking (MOT) and segmentation (MOTS) via an additional mask head. Furthermore, UniTrack [39] utilizes a shared appearance model and individual task heads, demonstrating that a single encoder can enable propagation and association across modalities. Additionally, UNICORN [10] explores learning robust representations by consolidating data across diverse tracking datasets and tasks.

2.2 Diffusion Models in Semantic Understanding

Generative models have recently been able to perform understanding tasks.

Representation Learning. Diffusion Autoencoders method [81] integrates a trainable encoder with a diffusion probabilistic model, thus forming a diffusion-based autoencoder framework. DRL [82] facilitates unsupervised representation learning by introducing an infinite-dimensional latent code that offers discretionary control over the granularity of detail. Furthermore, Asyrp [83] employs the asymmetric reverse process to explore and manipulate a semantic latent space, upholding the original performance, integrity, and consistency.

Visual Correspondence. DIFT [84] simulates the forward diffusion process, adding noise to input images and extracting features within the U-Net. Concurrently, Diffusion Hyperfeatures [85] uses feature aggregation and transforms intermediate feature maps from the diffusion process into a single, coherent descriptor map. Similarly, Zhang *et al*. [86] combines features from Stable Diffusion (SD) and DINOv2 [87] models, effectively merging the high-quality spatial information and capitalizing on both strengths. Additionally, Hedlin *et al*. [88] employs an off-the-shelf SD model to establish semantic correspondences between images by optimizing text embeddings to focus on specific regions.

Generative Perspectives in Object Tracking. A straightforward application of generative models in object tracking is to enrich and augment training data [89, 90]. QuoVadis [13] leverages the social generative adversarial network (GAN) [91] to sample multiple plausible future trajectories to account for the uncertainty in future positions. DiffusionTrack [12] utilizes the diffusion process in the bounding box decoder. Specifically, it pads prior *2D coordinate* bounding boxes with sampling noise, then transforms into tracking results via a denoising decoder.

These works utilize generative models to refine predictions on the coordinate space. We instead formulate the object tracking task entirely as a *purely visual generative process*, as illustrated in Fig. 2c and discussed in §2.3.

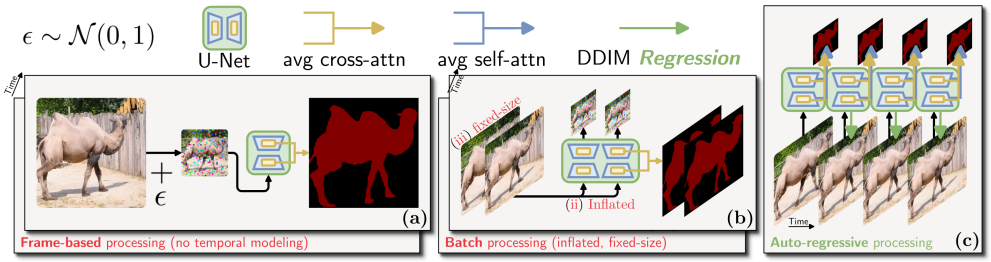

Fig. 3: Conceptual comparison between our DIFTracker and existing diffusion-based modeling. (a) Pseudo-noise latents [84], where sampled noise is directly added to the real image latent. (b) Inflated convolution and attention [44, 42], where trained weights are reshaped without fine-tuning. (c) Our proposed manipulated reconstruction, named ***Regression***, converts the tracking task into the temporal next-frame prediction task.

2.3 Discussion

This subsection discusses the following key aspects that distinguish our proposed paradigm from existing tracking methods, including a conceptual comparison to alternative approaches, and the fundamental innovative properties that enable our *Tracking-by-Diffusion* to stand out from the existing paradigms.

Concepts. Temporal modeling has posed challenges for applying diffusion models to lengthy video processing. Pseudo-noise approach [84] operates frame-based as in Fig. 3a and inflated video techniques [44, 43] rely on offline fixed-batch processing as in Fig. 3b, which cannot maintain temporal consistency flexibly. These concepts are further discussed in §5.4 to provide deeper insights into their intuition and performance. By reformulating the diffusion process to operate *online* and *auto-regressively* as briefly illustrated in Fig. 3c and detailed in Fig. 6, we offer flexibility for instance-based video understanding tasks. Our *Regression* modeling processes variable-length videos, which prior video diffusers [42, 43] have not achieved.

Additionally, although the conditioning mechanism $p_\theta(\mathbf{z}|\tau)$ of our DIFTracker is implemented by the attention mechanism [78] via Eqn. (7), it serves a different purpose compared to the *tracking-by-attention* paradigm. While *tracking-by-attention* approaches learn to match new detections to existing tracklets by representing them as transformer tokens, our transformer learns to condition the indication to the frame latents (**Remark 2**).

Unification. Existing approaches to the *tracking-by-unification* paradigm suffer from inefficient exploitation of available datasets. This is because single-target and multiple-target tasks are trained on separate branches [39] or different stages [30]. The discrepancy in architectures hinders fully leveraging the potential of labeled tracking instances. As a result, current hand-crafted techniques underperform compared to task-specific SOTA methods by a considerable margin.

In contrast, our proposed *Tracking-by-Diffusion* paradigm overcomes limitations in robustness. Our unified diffusion-based approach effortlessly handles point and pose regression, bounding box, segmentation, and referring tracking objectives while remaining dataset-agnostic via the Inversion-*Regression* process. This can be achieved

because the controlled injection of condition types during iterative diffusion allows unified tracking across indicator representations.

Application Coverage in Table 1 validates the unification advantages of our approach. As highlighted, the presented model DIFTracker supports unified tracking across *seven benchmarks* of *eight settings* comprising *five distinct categories of representation*. It handles single and multiple-target benchmarks, establishing a new standard in multiplicity, task flexibility, and novelty.

Noise Resilience. The proposed *Tracking-by-Diffusion* paradigm involves a reconstruction process from a noisy latent space. Alternative GAN frameworks also aim to find robust representation features [89] via a tamper-proof mechanism. However, perceptual compression via variational autoencoder (VAE) optimization exposes a vulnerability to accumulated noise interference [92]. Instead, diffusion gradually manipulating semantic concepts reduces compounding error by enabling step-wise reconstruction from pure noise [93, 12].

Bijective Representability. The bijective mapping [94] between the noise and data distributions allows a diffusion-based model to capture the salient visual feature of instances gradually. This key enables a straightforward manipulation of latent features for temporal modeling in our DIFTracker. Thus, our diffusion-based temporal modeling is *invertible* and *tractable* (see appendix B).

Temporal Reliance. We consider our approach can be similar to the propagation process [95], as it does not follow the common definition of *tracking-by-detection*, according to which detection and association are disentangled and sequential by conditioning the object detection stage on the tracking results from the previous timesteps. In contrast, *tracking-by-detection* is two-stage, commonly accepted in literature [95]: "First, objects are detected in each frame of the sequence and second, the detections are matched to form complete trajectories". From this definition, the association stage must associate the detected objects with old tracklets or use them to initialize new ones. In contrast, propagation addresses the training-inference gap through end-to-end learning from ordered video sequences, where detection is conditioned on previous timesteps. This approach intrinsically couples detection and propagation, unlike *tracking-by-detection* where the detector can be trained independently. However, this temporal reliance also presents the dependency of architecture on video data, not being able to train on image-level datasets nor benefit from large datasets like CrowdHuman [96]. Therefore, tracking performance is not our primary focus as other *tracking-by-detection* methods [97, 98], but instead is superior in indication representation as validated in Table 1.

3 Problem Formulation

Given two images $\mathbf{I}_t$ and $\mathbf{I}_{t+1}$ from a video sequence, and an indicator representation L_t (*e.g.* point, pose as structured point set, bounding box, segment or text) for an object in $\mathbf{I}_t$, we aim to find the respective region L_{t+1} in $\mathbf{I}_{t+1}$. The relationship between L_t and L_{t+1} can encode the temporal correspondence [99] (the localization of a deforming object over a video sequence), semantic correspondence [84, 85] (different objects with

similar semantic meanings), or geometric correspondence [100, 101] (the same object viewed from different viewpoints).

In this paper, we focus on *temporal correspondence*. We aim to establish matches between regions representing the same real-world object as it moves and potentially deforms or occludes 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. Along with the region L_t for localization, the ***auto-regressive*** *and* ***online*** *tracking objective* 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 distance [28] or softmax [102], and L is the decision variable in the optimization equation. The only special case is giving L_t as textual input and returning L_{t+1} as a bounding box or segment for the *referring object tracking* [103, 24] task. The objective is additionally illustrated in Fig. 4. We aim to explore how diffusion-based models can learn these temporal dynamics end-to-end to output consistent object representations frame-to-frame.

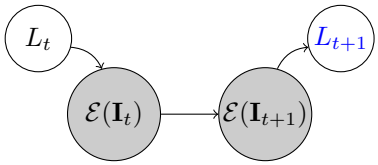

Fig. 4: Autoregressive objective.

In the next section, we will elaborate on how the auto-regressive objective can be achieved via the diffusion process to develop our proposed *Tracking-by-Diffusion* paradigm.

4 Methodology

4.1 Preliminaries

Latent Diffusion Models (LDMs) [3, 14] are introduced to denoise the latent space of an autoencoder. First, the encoder $\mathcal{E}(\cdot)$ compresses a RGB image $\mathbf{I}_t$ to an initial latent $\mathbf{z}_0 = \mathcal{E}(\mathbf{I}_t)$, which can be reconstructed back to image $\mathcal{D}(\mathbf{z}_0) \approx \mathbf{I}_t$. Second, an architecture with U-Net [104] blocks $\epsilon_\theta(\cdot,\cdot)$ containing cross-attention and self-attention [78] is trained to remove the artificial noise with the objective:

$$\min_{\theta} E_{\mathcal{E}(\mathbf{I}_t),\epsilon\sim\mathcal{N}(0,1),k\sim\text{Uniform}(1,T)}\left[\|\epsilon - \epsilon_\theta(\mathbf{z}_k, k)\|_2^2\right], \tag{2}$$

$\mathbf{z}_k$ is a noisy sample of $\mathbf{z}_0$ at step k, and $T = 50$ is the maximum denoising step.

To accelerate sampling, the *Denoising Diffusion Implicit Model* (DDIM) [4] is proposed as a more efficient class of iterative implicit probabilistic models with the same training procedure as denoising diffusion probabilistic models.

DDIM Inversion. Based on the ODE limit analysis [105] of the diffusion process, DDIM inversion [106], *i.e.*, *forward* or *sampling* process, is proposed to parameterize

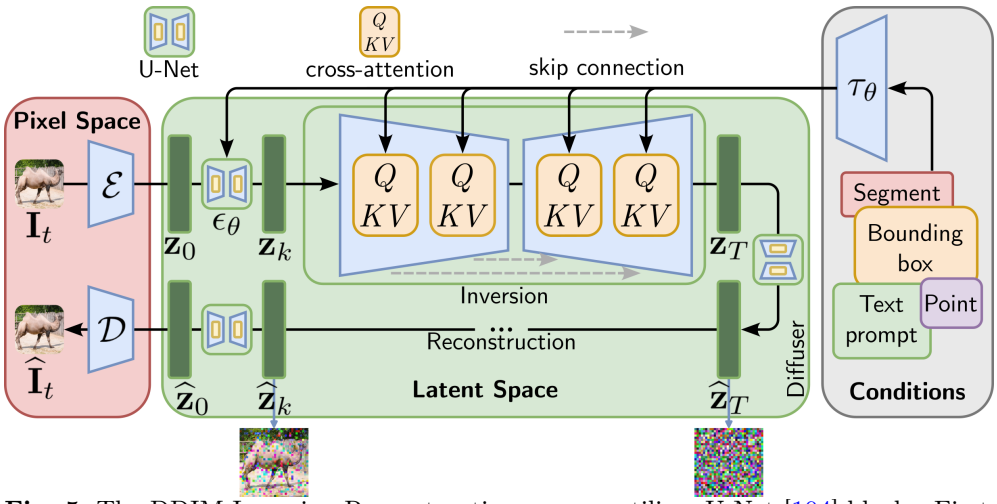

Fig. 5: The DDIM Inversion-Reconstruction process utilizes U-Net [104] blocks. First, a clean image $\mathbf{I}_t$ is converted to a noisy latent $\mathbf{z}_T$ via the Inversion process (top branch). Then, well-structured regions are reconstructed from that extremely noisy input via the Reconstruction process (bottom branch). Additionally, conditions can be added as indicators of the regions of interest. While the figure style is adapted from LDMs [14], we make two distinct changes reflecting our tracker: *parameterized* and *injected* Inversion (Eqn. (7)).

adding noise to a clean latent $\mathbf{z}_0$ in multiple steps $k \in [1, \dots, T]$:

$$\mathbf{z}_k = inv(\mathbf{z}_{k-1}) = \sqrt{\alpha_k}\,\frac{\mathbf{z}_{k-1} - \sqrt{1-\alpha_{k-1}}\,\epsilon_\theta(\mathbf{z}_{k-1}, k-1)}{\sqrt{\alpha_{k-1}}} + \sqrt{1-\alpha_k}\,\epsilon_\theta(\mathbf{z}_{k-1}, k-1), \tag{3}$$

where α_k is a parameter for noise scheduling, and $\epsilon_\theta(\mathbf{z}_{k-1}, k-1)$ is also the noise prediction respective to the score of the joint distribution.

DDIM Reconstruction. We denote $\widehat{\mathbf{z}}_k$ as the reconstructed latent at the k^{th} step, different from the clean latent notation $\mathbf{z}_k$. During inference, deterministic DDIM, *i.e.* *backward* or *denoising* process, is employed to convert the resulting random noise $\mathbf{z}_T$ in a sequence of reversed step $k \in [T, \dots, 1]$:

$$\widehat{\mathbf{z}}_{k-1} = rec(\widehat{\mathbf{z}}_k) = \sqrt{\alpha_{k-1}}\,\frac{\widehat{\mathbf{z}}_k - \sqrt{1-\alpha_k}\,\epsilon_\theta(\widehat{\mathbf{z}}_k, k)}{\sqrt{\alpha_k}} + \sqrt{1-\alpha_{k-1}}\,\epsilon_\theta(\widehat{\mathbf{z}}_k, k). \tag{4}$$

Such process shows the inverted latent $\mathbf{z}_T$ can be reconstructed to a latent $\widehat{\mathbf{z}}_0 = rec^{\times(T+1)}(\mathbf{z}_T) \approx \mathbf{z}_0$, one step longer than the Inversion.

These two processes, illustrated in Fig. 5, are the keys to developing the methodology for our *Tracking-by-Diffusion* paradigm. Inversion and Reconstruction utilize only a single network with U-Net blocks as the diffuser in these processes. In §4.2, we present

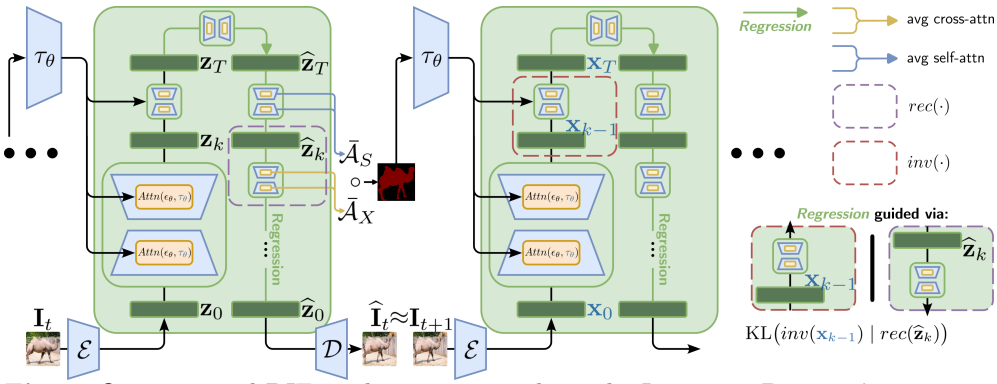

Fig. 6: Our proposed DIFTracker constructed via the Inversion-***Regression*** process w.r.t. $\mathbf{x}_{k-1}$ for temporal modeling. The current frame is input to the encoder $\mathcal{E}(\mathbf{I}_t)$ to produce an initial latent $\mathbf{z}_0$. The Inversion process $inv(\cdot)$ adds noise into the latent in a sequence of T steps. Next, our *Regression* process $reg(\cdot)$, which was originally the Reconstruction in the diffusion model, is manipulated through KL divergence optimization *w.r.t.* $\mathbf{x}_{k-1}$. 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 self-attention $\bar{\mathcal{A}}_S$ and cross-attention $\bar{\mathcal{A}}_X$ maps.

how to inject and extract internal states to identify the target regions holding coherence temporally, as in Fig. 6.

4.2 Tracking-by-Diffusion via DIFTracker

Remark 1. ***Powerful 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.*

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:

$$dist\Big(\mathcal{E}(\mathbf{I}_t), \mathcal{E}(\widehat{\mathbf{I}}_t)\Big) = 0 \ \textit{optimal} \ (\text{Eqn. } (2)), \text{ then } dist\Big(\mathcal{E}(\mathbf{I}_t)[L_t], \mathcal{E}(\widehat{\mathbf{I}}_t)[L_t]\Big) = 0. \quad (5)$$

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 $\mathbf{I}_t$ and the generated image $\widehat{\mathbf{I}}_t$.

Remark 2. ***Injected Inversion.*** *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 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. We unify all types of localization L_t, *e.g.* point, bounding box, segment, and especially text, as guided indicators. The diffusion objective from Eqn. (2) is now redefined to support injection as follows:

$$\min_{\theta} E_{\mathcal{E}(\mathbf{I}_t),\epsilon\sim\mathcal{N}(0,1),k\sim\text{Uniform}(1,T)}\Big[\big\|\epsilon-\epsilon_\theta\big(\mathbf{z}_k,k,\tau_\theta(L_t)\big)\big\|_2^2\Big], \tag{6}$$

where $\tau_\theta(\cdot)$ is the respective extractor, such as the Gaussian kernel for point, pooling layer for bounding box, and segment, or word embedding model for text. The injection is derived via the attention $Attn(\cdot,\cdot)$ [78] applied for U-Net ϵ_θ:

$$\epsilon_\theta\Big(\mathbf{z}_k,k,\tau_\theta(L_t)\Big)=\underbrace{\text{softmax}\Big(\frac{\epsilon_\theta(\mathbf{z}_k,k)\times W_Q\times\big(\tau_\theta(L_t)\times W_K\big)^\intercal}{\sqrt{d}}\Big)}_{Attn(\epsilon_\theta,\tau_\theta)}\times\big(\tau_\theta(L_t)\times W_V\big) \tag{7}$$

where $W_{[Q|K|V]}$ are projection matrices and d is the feature size. Here, we inject the indicators at the Inversion process, different from the original LDMs [14].

These two remarks support the diffusion process in capturing and semantically manipulating features for representing and distinguishing objects.

Remark 3. ***Temporal Modeling via Manipulated Reconstruction.*** *Since this approach aims to find correspondences as in Eqn. (1), the temporal modeling can be developed by maximizing the likelihood of reconstructing* $\widehat{\mathbf{I}}_t$ *toward* $\mathbf{I}_{t+1}$.

In layman's terms, we want the Reconstruction process to manipulate the resulting frame $\widehat{\mathbf{I}}_t$ to look similar to the next ***actual frame***:

$$\mathcal{D}(\widehat{\mathbf{z}}_0)=\widehat{\mathbf{I}}_t\approx\mathbf{I}_{t+1}. \tag{8}$$

Let us denote $\mathbf{x}_0=\mathcal{E}(\mathbf{I}_{t+1})$ and $\mathbf{x}_{k-1}=inv(\mathbf{x}_{k-2})$ are the initial and noisy latents of the next frame $\mathbf{I}_{t+1}$, which follows the Inversion process as in Eqn. (3). Assuming the pre-trained decoder $\mathcal{D}(\cdot)$ from LDMs can successfully reconstruct shapes, textures, and colors from the corresponding latent representation $\widehat{\mathbf{z}}_0$, the key optimization target is then the denoising process. To design that goal, ELBO, *i.e.* variational lower bound [107], is used to minimize the negative log-likelihood of $\widehat{\mathbf{z}}_{k-1}$ w.r.t. the desired latent $\mathbf{x}_{k-1}$. By reformulating the conditional probabilities via both processes $inv^{\times T}(\mathbf{x}_0)$ and $rec^{\times(T+1)}(\mathbf{z}_T)$, the ELBO is the KL divergence which can be written

explicitly is a sum of tractable losses at each time step k:

$$
\begin{aligned}
L_{\text{ELBO}} =& E_{\epsilon_\theta}\Big[\log \frac{inv^{\times T}(\mathbf{x}_0)}{rec^{\times (T+1)}(\mathbf{z}_T)}\Big] \\
=& E_{\epsilon_\theta}\Big[\log \frac{\prod_{k=1}^{T} inv(\mathbf{x}_{k-1})}{rec(\mathbf{z}_T)\prod_{k=1}^{T} rec(\widehat{\mathbf{z}}_k)}\Big] \\
=& E_{\epsilon_\theta}\Big[\log \frac{inv(\mathbf{x}_{T-1}) \times \prod_{k=2}^{T} inv(\mathbf{x}_{k-2})}{rec(\mathbf{z}_T) \times rec(\widehat{\mathbf{z}}_1) \times \prod_{k=2}^{T} rec(\widehat{\mathbf{z}}_k)}\Big] \\
=& E_{\epsilon_\theta}\Big[-\log rec(\widehat{\mathbf{z}}_1) + \sum_{k=2}^{T} \log \frac{inv(\mathbf{x}_{k-2})}{rec(\widehat{\mathbf{z}}_k)} + \log \frac{inv(\mathbf{x}_{T-1})}{rec(\mathbf{z}_T)}\Big] \\
=& E_{\epsilon_\theta}\Big[-\log \underbrace{rec(\widehat{\mathbf{z}}_1)}_{\widehat{\mathbf{z}}_0} + \sum_{k=2}^{T} \mathrm{KL}\big(\underbrace{inv(\mathbf{x}_{k-2})}_{\mathbf{x}_{k-1}} \| \underbrace{rec(\widehat{\mathbf{z}}_k)}_{\widehat{\mathbf{z}}_{k-1}}\big) + \mathrm{KL}\big(\underbrace{inv(\mathbf{x}_{T-1})}_{\mathbf{x}_T} \| \underbrace{rec(\mathbf{z}_T)}_{\widehat{\mathbf{z}}_T}\big)\Big] \\
=& \underbrace{-\log rec(\widehat{\mathbf{z}}_1)}_{\ell_0} + \sum_{k=2}^{T} \underbrace{\mathrm{KL}\big(inv(\mathbf{x}_{k-2}) \| rec(\widehat{\mathbf{z}}_k)\big)}_{\ell_{k-1}} + \underbrace{\mathrm{KL}\big(inv(\mathbf{x}_{T-1}) \| rec(\mathbf{z}_T)\big)}_{\ell_T} \\
=& \sum_{k=0}^{T} \ell_k
\end{aligned}
\tag{9}
$$

where we simplify the conditional probability notation $q(\mathbf{x}_{k-1}|\mathbf{x}_{k-2})$, Fig. 2a [3], as $\mathbf{x}_{k-1} = inv(\mathbf{x}_{k-2})$ and $p_\theta(\widehat{\mathbf{z}}_{k-1}|\widehat{\mathbf{z}}_k)$ as $\widehat{\mathbf{z}}_{k-1} = rec(\widehat{\mathbf{z}}_k)$ for the consistency in notations. Note that the one-shot finetuning here is to refine the reconstructed frame, not to learn the object identity or bounding box coordinates explicitly.

The lower bound of L_{ELBO} is $-\log rec(\widehat{\mathbf{z}}_1)$ via the proof below:

Proof.

$$
\begin{aligned}
-\log rec(\widehat{\mathbf{z}}_1) \leq& -\log rec(\widehat{\mathbf{z}}_1) + \mathrm{KL}\Big(inv^{\times T}(\mathbf{x}_0) \| rec^{\times T}(\mathbf{z}_T)\Big) \\
=& -\log rec(\widehat{\mathbf{z}}_1) + E_{\epsilon_\theta}\Big[\log \frac{inv^{\times T}(\mathbf{x}_0)}{rec^{\times T}(\mathbf{z}_T)}\Big] \\
=& -\log rec(\widehat{\mathbf{z}}_1) + E_{\epsilon_\theta}\Big[\log \frac{inv^{\times T}(\mathbf{x}_0) \times rec(\widehat{\mathbf{z}}_1)}{rec^{\times (T+1)}(\mathbf{z}_T)}\Big] \\
=& -\log rec(\widehat{\mathbf{z}}_1) + E_{\epsilon_\theta}\Big[\log \frac{inv^{\times T}(\mathbf{x}_0)}{rec^{\times (T+1)}(\mathbf{z}_T)} + \log rec(\widehat{\mathbf{z}}_1)\Big] \\
=& E_{\epsilon_\theta}\Big[\log \frac{inv^{\times T}(\mathbf{x}_0)}{rec^{\times (T+1)}(\mathbf{z}_T)}\Big] = L_{\text{ELBO}}
\end{aligned}
\tag{10}
$$

□

The resulting DDIM process, named *Regression*, which is not to be confused with the *tracking-by-regression* paradigm, but rather stemming from the *auto-**regressive*** property of our approach, is *manipulated w.r.t.* $\mathbf{x}_{k-1}$:

$$\begin{aligned}\mathbf{x}_{k-1} \approx \widehat{\mathbf{z}}_{k-1} &= reg(\widehat{\mathbf{z}}_k) \\ &= \sqrt{\alpha_{k-1}}\,\frac{\widehat{\mathbf{z}}_k - \sqrt{1-\alpha_k}\;\epsilon_\theta^{\mathbf{x}_{k-1}}(\widehat{\mathbf{z}}_k, k, \tau_\theta)}{\sqrt{\alpha_k}} + \sqrt{1-\alpha_{k-1}}\;\epsilon_\theta^{\mathbf{x}_{k-1}}(\widehat{\mathbf{z}}_k, k, \tau_\theta).\end{aligned} \tag{11}$$

This process operates sequentially for each new frame received. Thanks to the ELBO guiding the *Regression* $reg(\cdot)$ process, the distribution of instances can be efficiently captured. Moreover, this function is bijective, therefore, the *Regression* temporal modeling is exactable, *i.e.* $\widehat{\mathbf{I}}_t = \mathbf{I}_{t+1}$ (see §5.5 for empirical study and appendix B for theoretical validation).

Remark 4. ***Unified Head.*** *The attention maps* $Attn(\cdot,\cdot)$*, as illustrated in Fig. 2d, exhibit spatial correlations, can be utilized to identify the target locations.*

$\mathbf{I}_{t+1}$ *can be reconstructed from* $\mathbf{I}_t$, then internal attention is effortlessly extracted. To get into that, we first average the self- (denoted as $\bar{\mathcal{A}}_S$) and cross-attention maps (denoted as $\bar{\mathcal{A}}_X$) of the U-Net ϵ_θ over N layers and T timesteps:

$$\bar{\mathcal{A}}_S = \frac{1}{N\times T}\sum_{l=1}^{N}\sum_{k=1}^{T} Attn_{l,k}(\epsilon_\theta, \epsilon_\theta), \quad \bar{\mathcal{A}}_X = \frac{1}{N\times T}\sum_{l=1}^{N}\sum_{k=1}^{T} Attn_{l,k}(\epsilon_\theta, \tau_\theta). \tag{12}$$

The self-attention captures pairwise correlations among the indicators within the latent $\widehat{\mathbf{z}}_k$, which helps propagate the cross-attention to higher precise locations. Therefore, the element-wise product to enhance the resulting quality is operated:

$$\bar{\mathcal{A}}^* = \bar{\mathcal{A}}_S \circ \bar{\mathcal{A}}_X, \quad \bar{\mathcal{A}}^* \in [0,1]^{H\times W}, \quad \text{where } (H\times W) \text{ is the size of } \mathbf{I}_{t+1}. \tag{13}$$

Finally, for each type of the desired output, different mapping functions produce the prediction, illustrated in Fig. 2e, visualized in Fig. 7, and defined as:

$$L_{t+1} = \begin{cases} \arg\max(\bar{\mathcal{A}}^*), & \text{if point} \\ \bar{\mathcal{A}}^* > 0, & \text{if segment} \\ (\min_i \alpha, \min_j \alpha, \max_i \alpha, \max_j \alpha), \;\; \alpha = \left\{(i,j) \mid \bar{\mathcal{A}}^*_{i,j} > 0\right\}, & \text{if box} \end{cases} \tag{14}$$

5 Experimental Results

5.1 Benchmarks and Metrics

TAP-Vid [15] formalizes the problem of long-term physical **Point Tracking**. Three evaluation metrics are *Occlusion Accuracy (OA)*, $< \delta^x_{avg}$ averaging the position accuracy, and *Jaccard @* δ quantifying occlusion and position accuracies.

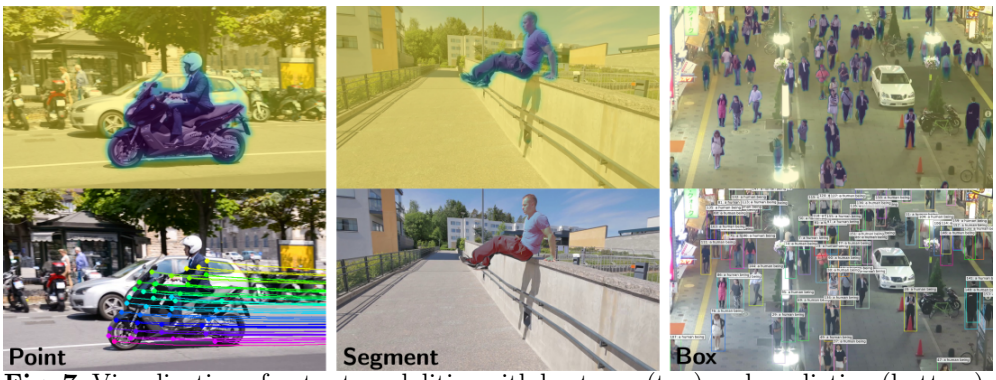

Fig. 7: Visualization of output modalities with heatmap (top) and prediction (bottom).

PoseTrack21 [17] is comparable to MOT17 [18]. It requires estimating both body **Pose** and **Bounding Box** for individuals, including joint visibility. Performance is assessed using keypoint-based mAP and standard metrics MOTA [108], IDF1 [109], and HOTA [110].

For **Segmentation Tracking** evaluation, DAVIS [19] and MOTS [21] datasets are used. Single-target assessment employs the Jaccard index $\mathcal{J}$ and contour accuracy $\mathcal{F}$, with an overall score calculated as their unweighted average [19] $\mathcal{J}\&\mathcal{F}$. Multiple-target evaluation uses MOTSA and MOTSP [21], which are analogous to MOTA and MOTP but use mask IoU instead of bounding box IoU for association measurement.

Referring Tracking is evaluated using LaSOT [23] and GroOT [24]. LaSOT measures Precision and Success metrics, while GroOT follows the MOT evaluation protocol with class-agnostic metrics.

5.2 Implementation Details

We finetune the DDIM *Regression* using a single-pass approach, similar to [42]. Unlike offline batch retraining, our technique updates the model in real-time as each new frame arrives. We base our work on LDM [14] for text-prompted scenarios and ADM [106] for localization, starting with their publicly available pre-trained models.

Table 2: Point tracking performance against several methods on TAP-Vid [15].

TAP-Vid	Kinetics [111]			Kubric [112]			DAVIS [19]			RGB-Stacking [113]		
	AJ	$< \delta^x_{avg}$	OA	AJ	$< \delta^x_{avg}$	OA	AJ	$< \delta^x_{avg}$	OA	AJ	$< \delta^x_{avg}$	OA
COTR [114]	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 [112]	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 [115]	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 [116]	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 [15]	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 [25]	57.1	70.0	87.6	84.3	**91.8**	**95.8**	59.8	72.3	87.6	**66.2**	77.4	**93.3**
DIFTracker	**57.7**	**72.3**	**89.2**	**85.4**	90.3	95.0	**62.1**	**74.5**	**88.8**	65.3	**77.6**	91.8
(i) *Added*	34.1	43.3	69.7	64.9	63.9	86.3	51.6	54.8	84.6	59.7	50.3	78.8
(ii) *Inflated*	53.6	64.3	88.5	80.5	86.4	87.0	62.0	66.9	84.9	62.3	71.0	89.6
(iii) *Fixed*	44.6	60.5	70.2	77.4	78.5	84.5	52.2	68.6	76.2	65.1	65.5	85.3

Table 3: Pose tracking performance against several methods on PoseTrack21 [17].

PoseTrack	mAP	MOTA	IDF1	HOTA	AssA	DetA	LocA
CorrTrack [117]	72.3	63.0	66.5	51.1	58.0	45.5	81.9
CorrTrack [117] w/ ReID	72.7	63.8	66.5	52.7	60.2	46.6	81.9
Tracktor++ [26] w/ poses	71.4	63.3	69.3	52.2	59.4	46.3	81.9
Tracktor++ [26] w/ corr.	73.6	61.6	69.3	54.1	54.1	44.6	81.2
DiffPose [118]	83.0	✗	✗	✗	✗	✗	✗
DIFTracker	**83.1**	**64.7**	**71.3**	**55.3**	**61.1**	**50.0**	**84.8**
(i) *Added*	69.1	43.6	55.1	40.7	42.2	39.3	81.5
(ii) *Inflated*	77.8	55.8	65.5	50.5	53.2	48.0	78.0
(iii) *Fixed*	77.2	52.8	63.6	43.2	46.3	40.3	70.5

We then refine the model using our proposed method, running 500 iterations with a learning rate of 3×10^{-5}. Training occurs on four NVIDIA Tesla A100 GPUs, processing one pair of frames per batch. We calculate the average of attention maps $\bar{\mathcal{A}}_S$ and $\bar{\mathcal{A}}_X$ during the first 80% of the 50 total DDIM steps.

5.3 Comparisons to the State-of-the-Arts

This section presents four main comparisons of our DIFTracker against *feature representation* and *temporal modeling* analogous SOTA methods.

Point Tracking. As presented in Table 2, our DIFTracker 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 Kinetics and DAVIS datasets (89.2 and 88.8 OA). RAFT [115] cannot easily detect occlusions and makes accumulated errors due to per-frame tracking. PIPs [116] and Tap-Net [15] lose flexibility by dividing the video into fixed-length segments. TAPIR [25] solely extracts local features around estimates rather than capturing the global context.

Pose Tracking. Table 3 compares our DIFTracker against other pose-tracking methods. We also include DiffPose [118], 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. Classic tracking methods, such as CorrTrack [117] and Tracktor++ [26], form appearance features with limited descriptiveness on keypoint representation. DiffPose [118] utilizes a similar generative process but operates on the heatmap domain rather than the pixel domain, which performs better than the former.

Table 4: Single object tracking without (left) and with (right) textual prompt input.

LaSOT	Precision	Success	Precision	Success
SiamRPN++ [119]	0.50	0.45	✗	✗
GlobalTrack [120]	0.53	0.52	✗	✗
OCEAN [121]	0.57	0.56	✗	✗
UNICORN [10]	**0.74**	0.68	✗	✗
GTI [27]	✗	✗	0.47	0.47
AdaSwitcher [122]	✗	✗	0.55	0.51
DIFTracker	**0.74**	**0.69**	**0.59**	**0.57**
(i) *Added*	0.52	0.49	0.46	0.45
(ii) *Inflated*	0.66	0.64	0.52	0.50
(iii) *Fixed*	0.54	0.55	0.49	0.48

Bounding Box Tracking. Table 4 shows the performance of single object tracking using bounding boxes or textual initialization. Similarly, Table 5 presents the performance of MOT using bounding boxes (left) or textual initialization (right). We compare our DIFTracker against MENDER [24] and a two-stage design, specifically MDETR [123] for the grounded detector, while TrackFormer [8] for the object tracker. Unlike these methods, which are limited to specific initialization types, our approach allows flexible injection inversion from any indicator type, improving unification capability. Moreover, capturing global contexts via the proposed DDIM *Regression* helps our model outperform methods relying solely on spatial contexts formulated via transformer-based learnable queries. Thus, our unified solution enables superior performance across diverse initialization types.

Segment Tracking. Finally, Table 6 presents our segment tracking performance against *unified* methods [39, 10], *single-target* methods [38, 125], and *multiple-target* methods [32, 7, 8, 126]. Our DIFTracker achieves the best sMOTSA of 67.4, an accurate object tracking and segmentation. Unified methods perform the task separately, either using different branches [39] or stages [10]. It leads to a discrepancy in networks. The unified DIFTracker avoids this shortcoming.

5.4 Ablation Study

This subsection is subdivided into three parts, each exploring an alternative to DIFTracker modeling. To substantiate the discussions, we include ablation studies in all tables below our base setting. These ablation settings are (i) *Added*, (ii) *Inflated*, and (iii) *Fixed*, elaborated below to discuss effectiveness.

Pseudo-noise Latents (i)**.** The real image $\mathbf{I}_t$ itself does not come from the training distribution of the U-Net ϵ_θ. DIFT [84] proposed a straightforward approximation as illustrated in Fig. 3a. Sampled noise respective to time step k is directly *added* to the real image latent $\mathbf{z}_k$. Without temporal modeling, this process approximately moves the image into the noise distribution that the U-Net was learned to reconstruct without fine-tuning, formally presented as follows:

$$\tilde{\mathbf{z}}_k = \epsilon_\theta(\mathbf{z}_0 + \epsilon, k), \quad \text{where } \epsilon \sim \mathcal{N}(0, 1). \tag{i}$$

It enables extracting latent features even though the real image does not match the training distribution. However, this approach could only partially bridge the distribution shift. As a result, this approach performs the worst performance overall as presented in Tables 2, 3, 4, 5, and 6.

Table 5: Multiple object tracking without (left) and with (right) textual prompt input.

MOT17-test	HOTA	IDF1	MOTA	MT	ML	IDs
FairMOT [124]	59.3	72.3	73.7	43.2%	17.3%	3303
ByteTrack [31]	63.1	77.3	**80.3**	53.2%	14.5%	2196
OCSORT [74]	63.2	77.5	78.0	41.0%	20.9%	**1950**
DiffusionTrack [12]	60.8	73.8	77.9	–	–	–
UNICORN [10]	61.7	77.2	75.5	**58.7%**	**11.2%**	5379
DIFTracker	**63.5**	**77.6**	78.0	54.2%	14.6%	4878

GroOT	MOTA	IDF1	HOTA	AssA	DetA	LocA
MDETR+TFm	62.6	64.7	51.5	50.9	52.2	81.1
MENDER [24]	65.5	63.4	66.7	52.9	52.9	81.3
DIFTracker	**68.8**	**68.3**	**57.3**	**56.8**	**58.1**	**82.2**
(i) *Added*	58.7	58.2	46.9	45.2	48.9	81.1
(ii) *Inflated*	63.0	58.6	48.4	48.0	49.1	81.1
(iii) *Fixed*	61.7	58.1	48.0	45.1	51.4	81.2

Table 6: Segment tracking performance on DAVIS [19] and MOTS [21].

VOS	$\mathcal{J}\&\mathcal{F}$	$\mathcal{J}$	$\mathcal{F}$
SiamMask [38]	56.4	54.3	58.5
Siam R-CNN [125]	70.6	66.1	75.0
UniTrack [39]	–	58.4	–
UNICORN [10]	69.2	65.2	73.2
DIFTracker	**75.7**	**72.7**	**78.6**
(i) *Added*	51.1	51.3	50.9
(ii) *Inflated*	68.5	67.8	69.1
(iii) *Fixed*	66.0	66.4	65.6

MOTS	sMOTSA	IDF1	MT	ML	IDSw
Track R-CNN [32]	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 [126]	62.3	42.9	56.7%	12.5%	541
UNICORN [10]	65.3	65.9	64.9%	10.1%	398
DIFTracker	**67.4**	**66.4**	**66.5%**	**8.5%**	484

Inflated Self-Attention (ii)**.** Instead of the one-shot fine-tuning strategy proposed in Eqn. (9), another approach can approximate this goal. To maintain the temporal coherence, VDMs [44] proposed to further extend the spatial 2D convolution layers and self-attention to the spatio-temporal domain as illustrated in Fig. 3b. Specifically, the *inflated* self-attention is derived as:

$$\text{from} \quad Attn\Big(\epsilon_\theta(\mathbf{z}_k,k),\epsilon_\theta(\mathbf{z}_k,k)\Big) \quad \text{to} \quad Attn\Big(\epsilon_\theta(\mathbf{z}_k,k),\epsilon_\theta\big([\mathbf{z}_k\|\mathbf{x}_k],k\big)\Big), \tag{ii}$$

where $[\cdot\|\cdot]$ is the concatenation operation, and the attention parameters only need to be reshaped without fine-tuning pre-trained weights. This solution is feasible for generating longer videos due to its flexibility. However, similar to (i), the actual distribution is not well captured, resulting in lower performance. Converting to this approach from our DIFTracker base leads to an approximate 5% performance drop as in Tables 2, 3, 4, 5, and 6. This decrease is anticipated as target distributions cannot fully be incorporated into the reconstruction process.

Semi-online Processing (iii)**.** In addition to frame-by-frame operation, we extend (ii) to a clip-by-clip paradigm as illustrated in Fig. 3b. The key motivation is validating the video diffusion models [42]. Unlike independent clip processing, we introduce cross-clip propagation for inter-clip object association. Formally, given a video clip $v_t \in \mathbb{R}^{I\times H\times W\times 3}$, where I is the *fixed* clip length (*i.e.* 16 frames) and t is the frame index, not the clip index (*i.e.* $v[t]$) $t \in [0, I-1]$, we pass it into the conditioned diffusion model. This semi-online approach extracts multiple frame features $\mathbf{z}_k^{v_t}$ via the U-Net $\epsilon_\theta(\cdot,\cdot)$. Here, a sparse causal attention computes matrices between frame $\mathbf{z}_k^{v_t}$ and two previous frames $\mathbf{z}_k^{v_0}$ and $\mathbf{z}_k^{v_{t-1}}$ as:

$$\text{from} \quad Attn\Big(\epsilon_\theta(\mathbf{z}_k^{v_t},k),\epsilon_\theta(\mathbf{z}_k^{v_t},k)\Big) \quad \text{to} \quad Attn\Big(\epsilon_\theta(\mathbf{z}_k^{v_t},k),\epsilon_\theta\big([\mathbf{z}_k^{v_0}\|\mathbf{z}_k^{v_{t-1}}],k\big)\Big). \tag{iii}$$

The outputs constitute I trajectory predictions across the I frames of the clip. For propagation, the last frame prediction is transferred to the subsequent clip during processing, as illustrated in Fig. 3b. This approach achieves mediocre performance, better than (i) but lower than (ii) because of the feature discrepancy between batches, as reported in Tables 2, 3, 4, 5, and 6.

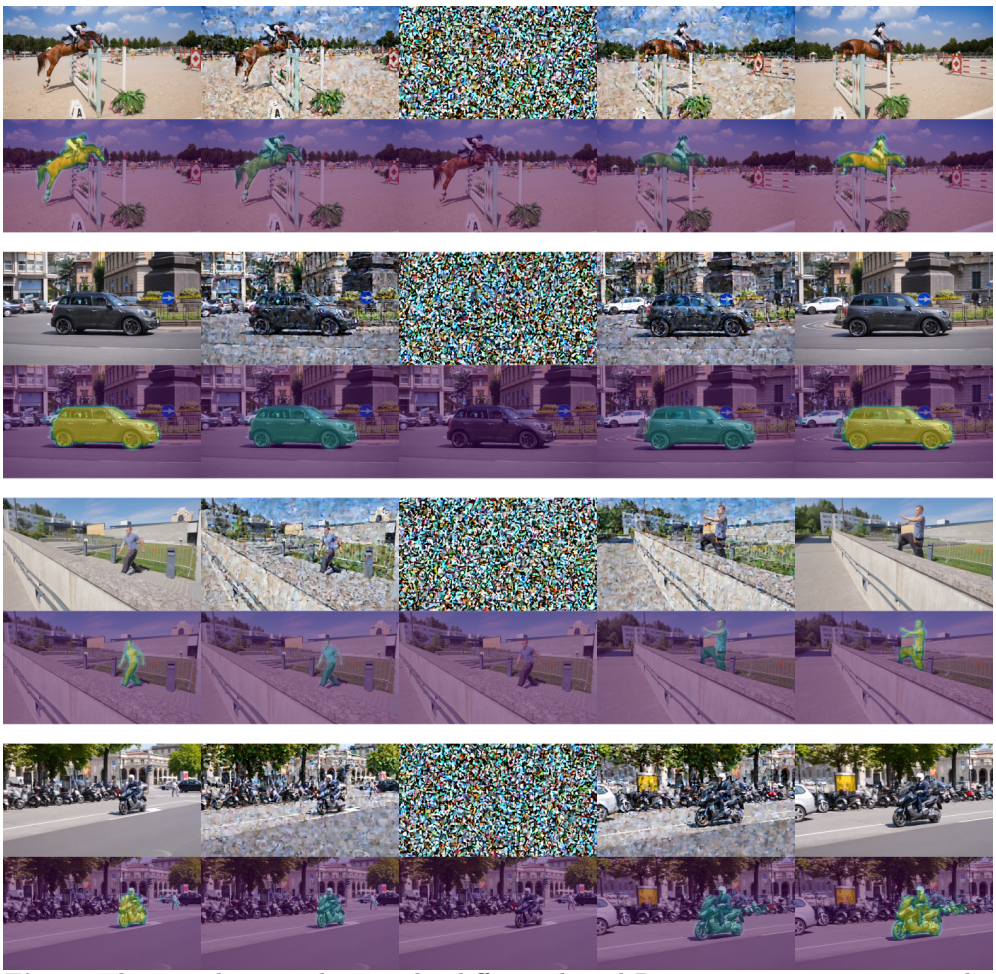

Fig. 8: The visualization depicts the diffusion-based Reconstruction process on the DAVIS benchmarks [19]. More results can be found in **Supplementary Video**.

5.5 Reconstruction Ability

Fig. 8 visualizes the proposed diffusion-based process utilized in our tracker framework. We construct the investigation examining the *Regression* quality $\widehat{\mathbf{I}}_t \approx \mathbf{I}_{t+1}$ to substantiate exactability performance via per-pixel MSE empirically. Additionally, we quantify the covariance between the per-pixel MSE gauging reconstruction quality and overall performance $\mathcal{J}\&\mathcal{F}$ on DAVIS [19] in Table 7. By timestep bound $T = 250$ in the reconstruction process, we can reconstruct an image extremely close to the original (with a per-pixel MSE of **0.04**). Our unified tracking framework establishes a single model capable of handling diverse modalities flexibly. Rather than expend effort on specialized models per tracking type, requiring extensive individual design and training

Table 7: Timestep bound T affects *Regression* quality, measured by MSE.

T	50	100	150	200	250
MSE ↓	20.56	15.46	10.32	5.18	**0.04**
$\mathcal{J}\&\mathcal{F}$ ↑	75.7	75.8	76.0	76.3	**76.5**
Regression 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

pipelines, our consolidation supports 5× current representations in one codebase. It reduces duplicated effort while enabling easy extendability as new modalities emerge.

Interpolation Operation. In light of improving the efficiency of the proposed paradigm, we reformulate the two-stage Inversion-*Regression* process to a single Interpolation process [127] and report in Table 7. With the base case $\widehat{\mathbf{x}}_T = \mathbf{z}_0^t$, the operation is formulated as an accumulative induction process:

$$k \in \{T-1, \dots, 1\},$$
$$\Big(\underbrace{\phi_\theta\big(\widehat{\mathbf{x}}_{k+1} + (\mathbf{x}_k - \mathbf{z}_{k+1}), k, \tau\big)}_{\widehat{\mathbf{x}}_k} \rightarrow \phi_\theta\big(\widehat{\mathbf{x}}_k \underbrace{+ (\mathbf{x}_{k-1} - \mathbf{z}_k)}_{\text{Interpolation operator}}, k-1, \tau\big)\Big). \tag{15}$$

where $\phi_\theta(\cdot)$ is a data interpolation model imposing temporal bias and has the same network structure of $\epsilon_\theta(\cdot)$ without changing layers. In layman's terms, the objective of the data interpolation 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 without the need for the Gaussian space projection step, as illustrated in Fig. 9. With the pre-trained decoder $\mathcal{D}(\cdot)$ in place, the key optimization target becomes the denoising process itself. More derivations, algorithms, and theoretical analyses of this process can be found in [127]. The reconstruction time is 3.2 seconds on a pair of frames at $T = 50$. The process utilizes an A100 GPU with at least 10GB of VRAM.

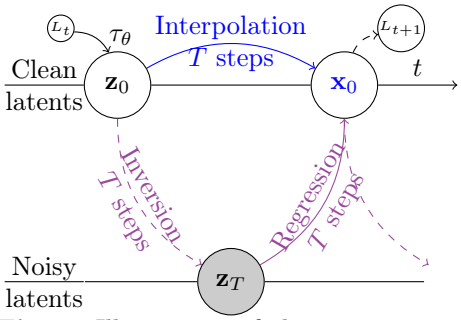

Fig. 9: Illustration of the regression and interpolation processes, where the purple dashed arrow is $q(\mathbf{z}_T|\mathbf{z}_0)$ and the purple solid arrow is $p_\epsilon(\mathbf{x}_0|\mathbf{z}_T)$, while the blue arrow illustrates $p_\phi(\mathbf{x}_0|\mathbf{z}_0^t)$.

6 Conclusion

We have presented a novel paradigm *Tracking-by-Diffusion*, which models the temporal correspondence of instances via the diffusion process. This paradigm has shown several innovative properties that stand out from existing paradigms, including

unification, broad application coverage, and bijective representability. The newly presented approach in this paradigm, DIFTracker, proposes to model the temporal domain by injecting the Inversion process and manipulating the Reconstruction process. By reformulating the diffusion process to operate online and auto-regressively (*i.e.* next-frame prediction), DIFTracker enables flexibility for instance-based video understanding, specifically object tracking. The proposed *Regression* modeling facilitates variable-length video processing, which previous video diffusion models have not achieved. The experiments show that our DIFTracker outperforms SOTA task-specific methods for representation learning and temporal modeling on seven tracking benchmarks covering five distinct categories.

Limitations. High resource utilization and time consumption due to the nature of diffusion models may limit the practicality of the LDMs. However, our unified approach allows us to handle multiple modalities rather than 5× efforts on specialized models.

Future Work. DIFTracker is a stepping stone towards more advanced *Tracking-by-Diffusion* tracking approaches in the future, especially to develop a foundation object tracking model or a new tracking approach that can manipulate visual contents [128] via the diffusion process. Exploring integrations between core representation and heuristic advances offers promising performance. Specific future directions include formulating new object representation [65] and management [129], motion [130, 6], state [131], temporal displacement [5], camera motion [75, 132, 133], geometric constrained [13], or open vocabulary [134] diffusion-based tracking approaches.

7 Data Availability Statement

The TAP-Vid dataset analyzed during the current study is available on GitHub. The dataset includes video and annotations, and it can be accessed at https://github.com/google-deepmind/tapnet. The data is published under the Creative Commons Attribution 4.0 International License.

The PoseTrack dataset analyzed during the current study is available on GitHub. The dataset includes video and annotations, and it can be accessed at https://github.com/anDoer/PoseTrack21. The data is published under the Creative Commons Attribution-NonCommercial 4.0 License.

The VOS dataset analyzed during the current study is available. The dataset includes video and annotations, and it can be accessed at https://youtube-vos.org/dataset/. The data is published under the Creative Commons Attribution 4.0 License.

The LaSOT dataset analyzed during the current study is available. The dataset includes video and annotations, and it can be accessed at http://vision.cs.stonybrook.edu/∼lasot/. The data is published under the Apache-2.0 License.

The MOT17, MOTS, and GroOT datasets analyzed during the current study are available in the MOT Challenge, an open-access data repository. The dataset includes video and annotations, and it can be accessed at https://motchallenge.net/. The data is published under the Creative Commons Attribution-NonCommercial-ShareAlike 3.0 License.

Please note that certain ethical and legal restrictions may apply to the data, and access may require compliance with applicable regulations and obtaining appropriate permissions.

References

[1] A. El-Nouby, M. Klein, S. Zhai, M.Á. Bautista, V. Shankar, A.T. Toshev, J.M. Susskind, A. Joulin, *Scalable Pre-training of Large Autoregressive Image Models*, in *Forty-first International Conference on Machine Learning* (2024)

[2] K. Tian, Y. Jiang, Z. Yuan, B. Peng, L. Wang, Visual autoregressive modeling: Scalable image generation via next-scale prediction. arXiv preprint arXiv:2404.02905 (2024)

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

[4] J. Song, C. Meng, S. Ermon, *Denoising Diffusion Implicit Models*, in *International Conference on Learning Representations* (2021)

[5] X. Zhou, V. Koltun, P. Krähenbühl, *Tracking Objects as Points*, in *Proceedings of the European Conference on Computer Vision (ECCV)* (2020), pp. 474–490

[6] N. Wojke, A. Bewley, D. Paulus, *Simple online and realtime tracking with a deep association metric*, in *2017 IEEE international conference on image processing (ICIP)* (IEEE, 2017), pp. 3645–3649

[7] J. Wu, J. Cao, L. Song, Y. Wang, M. Yang, J. Yuan, *Track to detect and segment: An online multi-object tracker*, in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (2021), pp. 12352–12361

[8] T. Meinhardt, A. Kirillov, L. Leal-Taixe, C. Feichtenhofer, *Trackformer: Multi-object tracking with transformers*, in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (2022), pp. 8844–8854

[9] F. Zeng, B. Dong, Y. Zhang, T. Wang, X. Zhang, Y. Wei, *Motr: End-to-end multiple-object tracking with transformer*, in *European Conference on Computer Vision* (Springer, 2022), pp. 659–675

[10] B. Yan, Y. Jiang, P. Sun, D. Wang, Z. Yuan, P. Luo, H. Lu, *Towards grand unification of object tracking*, in *European Conference on Computer Vision* (Springer, 2022), pp. 733–751

[11] S. Chen, P. Sun, Y. Song, P. Luo, *Diffusiondet: Diffusion model for object detection*, in *Proceedings of the IEEE/CVF International Conference on Computer Vision* (2023), pp. 19830–19843

[12] R. Luo, Z. Song, L. Ma, J. Wei, W. Yang, M. Yang, Diffusiontrack: Diffusion model for multi-object tracking. Proceedings of the AAAI Conference on Artificial Intelligence (2024)

[13] P. Dendorfer, V. Yugay, A. Ošep, L. Leal-Taixé, Quo vadis: Is trajectory forecasting the key towards long-term multi-object tracking? Advances in Neural Information Processing Systems **35** (2022)

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

[15] C. Doersch, A. Gupta, L. Markeeva, A. Recasens, L. Smaira, Y. Aytar, J. Carreira, A. Zisserman, Y. Yang, Tap-vid: A benchmark for tracking any point in a video. Advances in Neural Information Processing Systems **35**, 13610–13626 (2022)

[16] M. Andriluka, U. Iqbal, E. Insafutdinov, L. Pishchulin, A. Milan, J. Gall, B. Schiele, *Posetrack: A benchmark for human pose estimation and tracking*, in

Proceedings of the IEEE conference on computer vision and pattern recognition (2018), pp. 5167–5176

[17] A. Doering, D. Chen, S. Zhang, B. Schiele, J. 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* (2022), pp. 20963–20972

[18] A. Milan, L. Leal-Taixé, I. Reid, S. Roth, K. Schindler, MOT16: A benchmark for multi-object tracking. arXiv:1603.00831 [cs] (2016). ArXiv: 1603.00831

[19] F. Perazzi, J. Pont-Tuset, B. McWilliams, L. Van Gool, M. Gross, A. Sorkine-Hornung, *A benchmark dataset and evaluation methodology for video object segmentation*, in *Proceedings of the IEEE conference on computer vision and pattern recognition* (2016), pp. 724–732

[20] L. Yang, Y. Fan, N. Xu, *Video instance segmentation*, in *Proceedings of the IEEE/CVF International Conference on Computer Vision* (2019), pp. 5188–5197

[21] L. Porzi, M. Hofinger, I. Ruiz, J. Serrat, S.R. Bulo, P. Kontschieder, *Learning multi-object tracking and segmentation from automatic annotations*, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2020), pp. 6846–6855

[22] A. Geiger, P. Lenz, R. Urtasun, *Are we ready for autonomous driving? the kitti vision benchmark suite*, in *2012 IEEE conference on computer vision and pattern recognition* (IEEE, 2012), pp. 3354–3361

[23] H. Fan, H. Bai, L. Lin, F. Yang, P. Chu, G. Deng, S. Yu, Harshit, M. Huang, J. Liu, et al., Lasot: A high-quality large-scale single object tracking benchmark. International Journal of Computer Vision **129**, 439–461 (2021)

[24] P. Nguyen, K.G. Quach, K. Kitani, K. Luu, Type-to-track: Retrieve any object via prompt-based tracking. Advances in Neural Information Processing Systems **36** (2023)

[25] C. Doersch, Y. Yang, M. Vecerik, D. Gokay, A. Gupta, Y. Aytar, J. Carreira, A. Zisserman, Tapir: Tracking any point with per-frame initialization and temporal refinement. ICCV (2023)

[26] P. Bergmann, T. Meinhardt, L. Leal-Taixe, *Tracking without bells and whistles*, in *Proceedings of the IEEE/CVF International Conference on Computer Vision* (2019), pp. 941–951

[27] Z. Yang, T. Kumar, T. Chen, J. Su, J. Luo, Grounding-tracking-integration. IEEE Transactions on Circuits and Systems for Video Technology **31**(9), 3433–3443 (2020)

[28] N. Wojke, A. Bewley, *Deep cosine metric learning for person re-identification*, in *2018 IEEE winter conference on applications of computer vision (WACV)* (IEEE, 2018), pp. 748–756

[29] Y. Wang, K. Kitani, X. Weng, *Joint object detection and multi-object tracking with graph neural networks*, in *2021 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2021), pp. 13708–13715

[30] Z. Wang, L. Zheng, Y. Liu, Y. Li, S. Wang, *Towards real-time multi-object tracking*, in *Proceedings of the European Conference on Computer Vision (ECCV)* (Springer, 2020), pp. 107–122

[31] Y. Zhang, P. Sun, Y. Jiang, D. Yu, F. Weng, Z. Yuan, P. Luo, W. Liu, X. Wang, *ByteTrack: Multi-Object Tracking by Associating Every Detection Box*, in *Proceedings of the European Conference on Computer Vision (ECCV)* (2022)

[32] P. Voigtlaender, M. Krause, A. Osep, J. Luiten, B.B.G. Sekar, A. Geiger, B. Leibe, *Mots: Multi-object tracking and segmentation*, in *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (2019), pp. 7942–7951

[33] A. Ošep, W. Mehner, P. Voigtlaender, B. Leibe, *Track, then decide: Category-agnostic vision-based multi-object tracking*, in *2018 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2018), pp. 3494–3501

[34] Z. Xu, W. Zhang, X. Tan, W. Yang, H. Huang, S. Wen, E. Ding, L. 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* (Springer, 2020), pp. 264–281

[35] Y. Cui, T. Song, G. Wu, L. Wang, Mixformerv2: Efficient fully transformer tracking. Advances in Neural Information Processing Systems **36** (2024)

[36] R. Gao, L. Wang, *MeMOTR: Long-Term Memory-Augmented Transformer for Multi-Object Tracking*, in *Proceedings of the IEEE/CVF International Conference on Computer Vision* (2023), pp. 9901–9910

[37] H. Zhao, X. Wang, D. Wang, H. Lu, X. Ruan, Transformer vision-language tracking via proxy token guided cross-modal fusion. Pattern Recognition Letters (2023)

[38] Q. Wang, L. Zhang, L. Bertinetto, W. Hu, P.H. Torr, *Fast online object tracking and segmentation: A unifying approach*, in *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition* (2019), pp. 1328–1338

[39] Z. Wang, H. Zhao, Y.L. Li, S. Wang, P. Torr, L. Bertinetto, Do different tracking tasks require different appearance models? Advances in Neural Information Processing Systems **34**, 726–738 (2021)

[40] W. Lv, Y. Huang, N. Zhang, R.S. Lin, M. Han, D. 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)

[41] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, S. Ganguli, *Deep unsupervised learning using nonequilibrium thermodynamics*, in *International conference on machine learning* (PMLR, 2015), pp. 2256–2265

[42] J.Z. Wu, Y. Ge, X. Wang, S.W. Lei, Y. Gu, Y. Shi, W. Hsu, Y. Shan, X. Qie, M.Z. 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* (2023), pp. 7623–7633

[43] C. Qi, X. Cun, Y. Zhang, C. Lei, X. Wang, Y. Shan, Q. Chen, *Fatezero: Fusing attentions for zero-shot text-based video editing*, in *Proceedings of the IEEE/CVF International Conference on Computer Vision* (2023)

[44] J. Ho, T. Salimans, A. Gritsenko, W. Chan, M. Norouzi, D.J. Fleet, Video diffusion models. Advances in Neural Information Processing Systems **35**, 8633–8646 (2022)

[45] C. Feichtenhofer, A. Pinz, A. Zisserman, *Detect to track and track to detect*, in

Proceedings of the IEEE international conference on computer vision (2017), pp. 3038–3046

[46] Q. Liu, Q. Chu, B. Liu, N. Yu, *GSM: Graph Similarity Model for Multi-Object Tracking.*, in *IJCAI* (2020), pp. 530–536

[47] G. Brasó, L. Leal-Taixé, *Learning a neural solver for multiple object tracking*, in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (2020), pp. 6247–6257

[48] J. Berclaz, F. Fleuret, E. Turetken, P. Fua, Multiple object tracking using k-shortest paths optimization. IEEE transactions on pattern analysis and machine intelligence **33**(9), 1806–1819 (2011)

[49] H. Jiang, S. Fels, J.J. Little, *A linear programming approach for multiple object tracking*, in *2007 IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, 2007), pp. 1–8

[50] H. Pirsiavash, D. Ramanan, C.C. Fowlkes, *Globally-optimal greedy algorithms for tracking a variable number of objects*, in *CVPR 2011* (IEEE, 2011), pp. 1201–1208

[51] L. Zhang, Y. Li, R. Nevatia, *Global data association for multi-object tracking using network flows*, in *2008 IEEE conference on computer vision and pattern recognition* (IEEE, 2008), pp. 1–8

[52] H. Sheng, Y. Zhang, J. Chen, Z. Xiong, J. 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)

[53] C. Kim, F. Li, A. Ciptadi, J.M. Rehg, *Multiple hypothesis tracking revisited*, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (2015), pp. 4696–4704

[54] M. Keuper, S. Tang, B. Andres, T. Brox, B. Schiele, Motion segmentation & multiple object tracking by correlation co-clustering. IEEE transactions on pattern analysis and machine intelligence **42**(1), 140–153 (2018)

[55] Q. Yu, G. Medioni, I. Cohen, *Multiple target tracking using spatio-temporal markov chain monte carlo data association*, in *2007 IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, 2007), pp. 1–8

[56] S. Tang, M. Andriluka, B. Andres, B. Schiele, *Multiple people tracking by lifted multicut and person re-identification*, in *Proceedings of the IEEE conference on computer vision and pattern recognition* (2017), pp. 3539–3548

[57] D.M. Nguyen, R. Henschel, B. Rosenhahn, D. Sonntag, P. 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* (2022), pp. 8866–8875

[58] R. Henschel, L. Leal-Taixé, D. Cremers, B. Rosenhahn, Improvements to frank-wolfe optimization for multi-detector multi-object tracking. arXiv preprint arXiv:1705.08314 **8** (2017)

[59] A. Hornakova, R. Henschel, B. Rosenhahn, P. Swoboda, *Lifted disjoint paths with application in multiple object tracking*, in *International Conference on Machine Learning* (PMLR, 2020), pp. 4364–4375

[60] K.G. Quach, P. Nguyen, H. Le, T.D. Truong, C.N. Duong, M.T. Tran, K. 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* (2021), pp. 13784–13793

[61] L. Leal-Taixé, C. Canton-Ferrer, K. Schindler, *Learning by tracking: Siamese CNN for robust target association*, in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops* (2016), pp. 33–40

[62] J. Pang, L. Qiu, X. Li, H. Chen, Q. Li, T. Darrell, F. Yu, *Quasi-dense similarity learning for multiple object tracking*, in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (2021), pp. 164–173

[63] E. Ristani, C. Tomasi, *Features for multi-target multi-camera tracking and re-identification*, in *Proceedings of the IEEE conference on computer vision and pattern recognition* (2018), pp. 6036–6046

[64] P. Chu, H. 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* (2019), pp. 6172–6181

[65] J. Rajasegaran, G. Pavlakos, A. Kanazawa, J. Malik, *Tracking People by Predicting 3D Appearance, Location and Pose*, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2022), pp. 2740–2749

[66] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, S. Savarese, *Social lstm: Human trajectory prediction in crowded spaces*, in *Proceedings of the IEEE conference on computer vision and pattern recognition* (2016), pp. 961–971

[67] L. Leal-Taixé, G. Pons-Moll, B. 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)* (IEEE, 2011), pp. 120–127

[68] A. Robicquet, A. Sadeghian, A. Alahi, S. Savarese, *Learning social etiquette: Human trajectory prediction in crowded scenes*, in *European Conference on Computer Vision (ECCV)*, vol. 2 (2016), p. 5

[69] A. Andriyenko, K. Schindler, *Multi-target tracking by continuous energy minimization*, in *CVPR 2011* (IEEE, 2011), pp. 1265–1272

[70] L. Chen, H. Ai, Z. Zhuang, C. 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)* (IEEE, 2018), pp. 1–6

[71] S. Pellegrini, A. Ess, K. Schindler, L. Van Gool, *You'll never walk alone: Modeling social behavior for multi-target tracking*, in *2009 IEEE 12th international conference on computer vision* (IEEE, 2009), pp. 261–268

[72] P. Scovanner, M.F. Tappen, *Learning pedestrian dynamics from the real world*, in *2009 IEEE 12th International Conference on Computer Vision* (IEEE, 2009), pp. 381–388

[73] K. Yamaguchi, A.C. Berg, L.E. Ortiz, T.L. Berg, *Who are you with and where are you going?*, in *CVPR 2011* (IEEE, 2011), pp. 1345–1352

[74] J. Cao, J. Pang, X. Weng, R. Khirodkar, K. Kitani, *Observation-centric sort: Rethinking sort for robust multi-object tracking*, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2023), pp. 9686–9696

[75] N. Aharon, R. Orfaig, B.Z. Bobrovsky, Bot-sort: Robust associations multi-pedestrian tracking. arXiv preprint arXiv:2206.14651 (2022)

[76] L. Leal-Taixé, M. Fenzi, A. Kuznetsova, B. Rosenhahn, S. Savarese, *Learning an image-based motion context for multiple people tracking*, in *Proceedings of the IEEE conference on computer vision and pattern recognition* (2014), pp. 3542–3549

[77] P. Tokmakov, J. Li, W. Burgard, A. Gaidon, *Learning to track with object permanence*, in *Proceedings of the IEEE/CVF International Conference on Computer Vision* (2021), pp. 10860–10869

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

[79] Y. Zhang, T. Wang, X. 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* (2023), pp. 22056–22065

[80] J. Cai, M. Xu, W. Li, Y. Xiong, W. Xia, Z. Tu, S. Soatto, *Memot: Multi-object tracking with memory*, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2022), pp. 8090–8100

[81] K. Preechakul, N. Chatthee, S. Wizadwongsa, S. Suwajanakorn, *Diffusion autoencoders: Toward a meaningful and decodable representation*, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2022), pp. 10619–10629

[82] S. Mittal, K. Abstreiter, S. Bauer, B. Schölkopf, A. Mehrjou, *Diffusion based representation learning*, in *International Conference on Machine Learning* (PMLR, 2023), pp. 24963–24982

[83] M. Kwon, J. Jeong, Y. Uh, *Diffusion Models Already Have A Semantic Latent Space*, in *The Eleventh International Conference on Learning Representations* (2023)

[84] L. Tang, M. Jia, Q. Wang, C.P. Phoo, B. Hariharan, *Emergent Correspondence from Image Diffusion*, in *Thirty-seventh Conference on Neural Information Processing Systems* (2023)

[85] G. Luo, L. Dunlap, D.H. Park, A. Holynski, T. Darrell, *Diffusion Hyperfeatures: Searching Through Time and Space for Semantic Correspondence*, in *Advances in Neural Information Processing Systems*, vol. 36 (2023)

[86] J. Zhang, C. Herrmann, J. Hur, L.P. Cabrera, V. Jampani, D. Sun, M.H. Yang, A tale of two features: Stable diffusion complements dino for zero-shot semantic correspondence. Advances in Neural Information Processing Systems **36** (2023)

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

[88] E. Hedlin, G. Sharma, S. Mahajan, H. Isack, A. Kar, A. Tagliasacchi, K.M. Yi, Unsupervised semantic correspondence using stable diffusion. Advances in Neural Information Processing Systems **36** (2023)

[89] C.D. Prakash, L.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)
[90] P. Li, Z. Liu, K. Chen, L. Hong, Y. Zhuge, D.Y. Yeung, H. Lu, X. Jia, Trackdiffusion: Multi-object tracking data generation via diffusion models. arXiv preprint arXiv:2312.00651 (2023)
[91] A. Gupta, J. Johnson, L. Fei-Fei, S. Savarese, A. Alahi, *Social gan: Socially acceptable trajectories with generative adversarial networks*, in *Proceedings of the IEEE conference on computer vision and pattern recognition* (2018), pp. 2255–2264
[92] H. Fang, G. Carbajal, S. Wermter, T. Gerkmann, *Variational autoencoder for speech enhancement with a noise-aware encoder*, in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (IEEE, 2021), pp. 676–680
[93] Y. Ye, K. Xu, Y. Huang, R. Yi, Z. Cai, *DiffusionEdge: Diffusion Probabilistic Model for Crisp Edge Detection*, in *Proceedings of the AAAI Conference on Artificial Intelligence* (2024)
[94] Z. Kong, W. Ping, *On Fast Sampling of Diffusion Probabilistic Models*, in *ICML Workshop on Invertible Neural Networks, Normalizing Flows, and Explicit Likelihood Models* (2021)
[95] L. Leal-Taixe, Multiple object tracking with context awareness. arxiv preprint 1411.7935 (2014)
[96] S. Shao, Z. Zhao, B. Li, T. Xiao, G. Yu, X. Zhang, J. Sun, Crowdhuman: A benchmark for detecting human in a crowd. arXiv preprint arXiv:1805.00123 (2018)
[97] F. Yang, Z. Wang, Y. Wu, S. Sakti, S. Nakamura, Tackling multiple object tracking with complicated motions—re-designing the integration of motion and appearance. Image and Vision Computing **124**, 104514 (2022)
[98] S. Wang, H. Sheng, D. Yang, Y. Zhang, Y. Wu, S. Wang, Extendable multiple nodes recurrent tracking framework with rtu++. IEEE Transactions on Image Processing **31**, 5257–5271 (2022)
[99] N. Karaev, I. Rocco, B. Graham, N. Neverova, A. Vedaldi, C. Rupprecht, Cotracker: It is better to track together. arXiv preprint arXiv:2307.07635 (2023)
[100] T.D. Truong, C.N. Duong, A. Dowling, S.L. Phung, J. Cothren, K. Luu, Crovia: Seeing drone scenes from car perspective via cross-view adaptation. arXiv preprint arXiv:2304.07199 (2023)
[101] P. Nguyen, K.G. Quach, J. Gauch, S.U. Khan, B. Raj, K. Luu, Utopia: Unconstrained tracking objects without preliminary examination via cross-domain adaptation. arXiv preprint arXiv:2306.09613 (2023)
[102] T. Fischer, T.E. Huang, J. Pang, L. Qiu, H. Chen, T. Darrell, F. Yu, Qdtrack: Quasi-dense similarity learning for appearance-only multiple object tracking. IEEE Transactions on Pattern Analysis and Machine Intelligence (2023)
[103] D. Wu, W. Han, T. Wang, X. Dong, X. Zhang, J. Shen, *Referring Multi-Object Tracking*, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2023), pp. 14633–14642
[104] O. Ronneberger, P. Fischer, T. 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 (Springer, 2015), pp. 234–241

[105] C. Lu, Y. Zhou, F. Bao, J. Chen, C. Li, J. Zhu, Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps. Advances in Neural Information Processing Systems **35**, 5775–5787 (2022)

[106] P. Dhariwal, A. Nichol, Diffusion models beat gans on image synthesis. Neural Information Processing Systems (2021)

[107] D.P. Kingma, M. Welling, Auto-encoding variational bayes. International Conference on Learning Representations (2014)

[108] K. Bernardin, R. Stiefelhagen, Evaluating multiple object tracking performance: the clear mot metrics. EURASIP Journal on Image and Video Processing **2008**, 1–10 (2008)

[109] E. Ristani, F. Solera, R. Zou, R. Cucchiara, C. Tomasi, *Performance measures and a data set for multi-target, multi-camera tracking*, in *European conference on computer vision* (Springer, 2016), pp. 17–35

[110] J. Luiten, A. Osep, P. Dendorfer, P. Torr, A. Geiger, L. Leal-Taixé, B. Leibe, Hota: A higher order metric for evaluating multi-object tracking. International journal of computer vision **129**, 548–578 (2021)

[111] J. Carreira, A. Zisserman, *Quo vadis, action recognition? a new model and the kinetics dataset*, in *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2017), pp. 6299–6308

[112] K. Greff, F. Belletti, L. Beyer, C. Doersch, Y. Du, D. Duckworth, D.J. Fleet, D. Gnanapragasam, F. Golemo, C. Herrmann, et al., *Kubric: A scalable dataset generator*, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2022), pp. 3749–3761

[113] A.X. Lee, C.M. Devin, Y. Zhou, T. Lampe, K. Bousmalis, J.T. Springenberg, A. Byravan, A. Abdolmaleki, N. Gileadi, D. Khosid, et al., *Beyond pick-and-place: Tackling robotic stacking of diverse shapes*, in *Conference on Robot Learning* (PMLR, 2022), pp. 1089–1131

[114] W. Jiang, E. Trulls, J. Hosang, A. Tagliasacchi, K.M. Yi, *Cotr: Correspondence transformer for matching across images*, in *Proceedings of the IEEE/CVF International Conference on Computer Vision* (2021), pp. 6207–6217

[115] Z. Teed, J. 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* (Springer, 2020), pp. 402–419

[116] A.W. Harley, Z. Fang, K. Fragkiadaki, *Particle video revisited: Tracking through occlusions using point trajectories*, in *European Conference on Computer Vision* (Springer, 2022), pp. 59–75

[117] U. Rafi, A. Doering, B. Leibe, J. 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* (Springer, 2020), pp. 36–52

[118] R. Feng, Y. Gao, T.H.E. Tse, X. Ma, H.J. Chang, *DiffPose: SpatioTemporal diffusion model for video-based human pose estimation*, in *Proceedings of the IEEE/CVF International Conference on Computer Vision* (2023), pp. 14861–14872

[119] B. Li, J. Yan, W. Wu, Z. Zhu, X. Hu, *High performance visual tracking with siamese region proposal network*, in *Proceedings of the IEEE conference on computer vision and pattern recognition* (2018), pp. 8971–8980

[120] L. Huang, X. Zhao, K. Huang, *Globaltrack: A simple and strong baseline for long-term tracking*, in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34 (2020), pp. 11037–11044

[121] Z. Zhang, H. Peng, J. Fu, B. Li, W. 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* (Springer, 2020), pp. 771–787

[122] X. Wang, X. Shu, Z. Zhang, B. Jiang, Y. Wang, Y. Tian, F. 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* (2021), pp. 13763–13773

[123] A. Kamath, M. Singh, Y. LeCun, G. Synnaeve, I. Misra, N. Carion, *Mdetr-modulated detection for end-to-end multi-modal understanding*, in *Proceedings of the IEEE/CVF International Conference on Computer Vision* (2021), pp. 1780–1790

[124] Y. Zhang, C. Wang, X. Wang, W. Zeng, W. Liu, Fairmot: On the fairness of detection and re-identification in multiple object tracking. International journal of computer vision **129**, 3069–3087 (2021)

[125] P. Voigtlaender, J. Luiten, P.H. Torr, B. Leibe, *Siam r-cnn: Visual tracking by re-detection*, in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (2020), pp. 6578–6588

[126] Z. Xu, W. Yang, W. Zhang, X. Tan, H. Huang, L. 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)

[127] P. Nguyen, N.H. Le, J. Cothren, A. Yilmaz, K. Luu, *DINTR: Tracking via Diffusion-based Interpolation*, in *The Thirty-eighth Annual Conference on Neural Information Processing Systems* (2024). URL https://openreview.net/forum?id=gAgwqHOBIg

[128] U. Singer, A. Polyak, T. Hayes, X. Yin, J. An, S. Zhang, Q. Hu, H. Yang, O. Ashual, O. Gafni, D. Parikh, S. Gupta, Y. Taigman, *Make-A-Video: Text-to-Video Generation without Text-Video Data*, in *The Eleventh International Conference on Learning Representations* (2023)

[129] D. Stadler, J. Beyerer, *Improving multiple pedestrian tracking by track management and occlusion handling*, in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (2021), pp. 10958–10967

[130] A. Bewley, Z. Ge, L. Ott, F. Ramos, B. Upcroft, *Simple online and realtime tracking*, in *2016 IEEE International Conference on Image Processing (ICIP)* (IEEE, 2016), pp. 3464–3468

[131] S. Sun, N. Akhtar, X. Song, H. Song, A. Mian, M. Shah, *Simultaneous Detection and Tracking with Motion Modelling for Multiple Object Tracking*, in *Proceedings of the European Conference on Computer Vision (ECCV)* (2020), pp. 626–643

[132] P. Nguyen, K.G. Quach, C.N. Duong, N. Le, X.B. Nguyen, K. 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* (2022), pp. 2569–2578

[133] P. Nguyen, K.G. Quach, C.N. Duong, S.L. Phung, N. Le, K. Luu, Multi-camera multi-object tracking on the move via single-stage global association approach. Pattern Recognition p. 110457 (2024)

[134] S. Li, T. Fischer, L. Ke, H. Ding, M. Danelljan, F. Yu, *OVTrack: Open-Vocabulary Multiple Object Tracking*, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2023), pp. 5567–5577

[135] L. Dinh, D. Krueger, Y. Bengio, Nice: Non-linear independent components estimation. arXiv preprint arXiv:1410.8516 (2014)

[136] L. Dinh, J. Sohl-Dickstein, S. Bengio, *Density estimation using Real NVP*, in *International Conference on Learning Representations* (2017)

[137] R. Durrett, *Probability: theory and examples*, vol. 49 (Cambridge university press, 2019)

[138] B.D. Anderson, Reverse-time diffusion equation models. Stochastic Processes and their Applications **12**(3), 313–326 (1982)

[139] B. Oksendal, *Stochastic differential equations: an introduction with applications* (Springer Science & Business Media, 2013)

[140] Y. Song, J. Sohl-Dickstein, D.P. Kingma, A. Kumar, S. Ermon, B. Poole, *Score-Based Generative Modeling through Stochastic Differential Equations*, in *International Conference on Learning Representations* (2021)

Appendices

A Notations

Table A.8: Notations used throughout the paper.

	Main Formulation
$\mathbf{I}_t$	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
$\mathcal{D}(\widehat{\mathbf{z}}_0)$	Visual decoder decoding latent feature to image
θ	Deep network parameters
ϵ	A noise variable, $\epsilon \sim \mathcal{N}(0, 1)$
$\epsilon_\theta(\mathbf{z}_k, k)$	Denoising autoencoders, *i.e.*, U-Net blocks
$\|\cdot\|_2^2$	L^2 norm
$\mathbf{z}_0, \ldots, \mathbf{z}_k, \ldots, \mathbf{z}_T$	Latent variables encoding $\mathbf{I}_t$ in the Inversion process
$\widehat{\mathbf{z}}_0, \ldots, \widehat{\mathbf{z}}_k, \ldots, \widehat{\mathbf{z}}_T$	Latent variables $\mathbf{I}_t$ in Reconstruction/Regression process
$\mathbf{x}_0, \ldots, \mathbf{x}_k, \ldots, \mathbf{x}_T$	Latent variables encoding $\mathbf{I}_{t+1}$ in the Inversion process
α_k, α_{k-1}	Noise scheduling parameters
$inv^{\times T}(\cdot)$	Inversion process repeats T times
$rec^{\times (T+1)}(\cdot)$	Reconstruction process repeats $T+1$ times
$\tau_\theta(\cdot)$	Indicator feature extractor
$E_{\epsilon_\theta} L(\cdot)$	Expectation of a loss function $L(\cdot)$ with respect to ϵ_θ
$\mathrm{KL}(P\|Q)$	Kullback-Leibler divergence of P and Q
$q(\mathbf{x}_k\|\mathbf{x}_{k-1})$	Conditional probability of $\mathbf{x}_k$ given $\mathbf{x}_{k-1}$, $q(\mathbf{x}_k\|\mathbf{x}_{k-1}) \equiv inv(\mathbf{x}_{k-1})$

$p_\theta(\widehat{\mathbf{z}}_{k-1}\|\widehat{\mathbf{z}}_k)$	Parameterized conditional probability of $\widehat{\mathbf{z}}_{k-1}$ given $\widehat{\mathbf{z}}_k$, $p_\theta(\widehat{\mathbf{z}}_{k-1}\|\widehat{\mathbf{z}}_k) \equiv rec(\widehat{\mathbf{z}}_k)$
$\bar{\mathcal{A}}_S$	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

Inflated Approaches

$\bar{\mathbf{z}}_k$	Pseudo-noise latent variable at the k^{th} step
$[\,\cdot\,\|\|\,\cdot\,]$	Concatenation operation
$\mathbf{z}_k^{v_0}, \ldots, \mathbf{z}_k^{v_{t-1}}, \mathbf{z}_k^{v_t}$	The k^{th}-step latent variables of different frames in a clip, $\mathbf{z}_k^{v_{t+1}} \equiv \mathbf{x}_k$

B Mathematical Details

Bijectivity and Tractability. Prior works have highlighted limitations in flow-based generative models [135, 136], including constrained network architectures and difficulty scaling training. However, an emerging connection between diffusion models and continuous normalizing flows may overcome these barriers. We analyze recent developments leveraging the probability flow ordinary differential equation (ODE) for scalable and stable sampling.

Formally, the stochastic differential equation (SDE) in diffusion models induces intermediate distributions evolving over time, indexed by k:

$$\mathrm{d}\mathbf{x} = \mathbf{f}(\mathbf{x}, k)\mathrm{d}k + g(k)\mathrm{d}\mathbf{w}. \tag{B.16}$$

This equation establishes a connection between the incremental alteration in $\mathbf{x}$ and a corresponding infinitesimal variation in k. The term $\mathrm{d}\mathbf{w}$ denotes infinitesimal Gaussian noise, commonly referred to as the Wiener process [137]. The function $\mathbf{f}(\cdot,\cdot)$ and $g(\cdot)$ are designated as the drift and diffusion coefficients, respectively. Specific selections for $\mathbf{f}(\cdot,\cdot)$ and $g(\cdot)$ lead to continuous-time representations of the Markov chains employed in the formulation of DDPMs.

Another SDE that describes the process in the other direction, *i.e.* reverses time [138], can be derived as follows:

$$\mathrm{d}\mathbf{x} = \Big(\mathbf{f}(\mathbf{x}, k) - g(k)^2\nabla_{\mathbf{x}} \log p_k(\mathbf{x})\Big)\mathrm{d}k + g(k)\mathrm{d}\bar{\mathbf{w}}. \tag{B.17}$$

The Fokker-Planck equation [139] governs these transitions and evolutions [140]. Notably, an ODE exists describing a deterministic process with identical time-dependent

distributions as the SDE:

$$\mathrm{d}\mathbf{x} = \Big(\mathbf{f}(\mathbf{x}, k) - \frac{1}{2} g(k)^2 \nabla_{\mathbf{x}} \log p_k(\mathbf{x})\Big)\mathrm{d}k. \tag{B.18}$$

This probability flow ODE enables ***bijective*** mapping between base and data distributions through *forward* and *backward* ODE simulation (*i.e.* Inversion and Reconstruction).

Crucially, the deterministic sampling process avoids optimization difficulties with the stochastic SDE, while matching distributions. The induced continuous normalizing flow facilitates latent vector manipulation and data-latent bidirectional mapping. Additionally, the ODE supports ***tractable*** likelihood computation via the ELBO, previously intractable for diffusion models.

Leveraging the properties of ***bijectivity*** and ***tractability***, we can establish a mapping between data points and their associated latent representations. This involves simulating the ODE forward, manipulating the latent representations, and subsequently mapping them precisely back to the data space. *A sufficiently high timestep bound* T, of $k \in [1, \ldots, T]$, guarantees the simulation of the ODE in reverse, ensuring the ***exactability*** in the Reconstruction/*Regression* process as reported in §5.5.

C 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:

$$\mathcal{T} = \Big[\tau_\theta(L_t^0) \| \ldots \| \tau_\theta(L_t^i) \| \ldots \| \tau_\theta(L_t^{M-1})\Big]. \tag{C.19}$$

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

$$\bar{\mathcal{A}}_X = [\bar{\mathcal{A}}_X^0 \| \ldots \| \bar{\mathcal{A}}_X^i \| \ldots \| \bar{\mathcal{A}}_X^{M-1}], \quad \bar{\mathcal{A}}_X \in [0, 1]^{M \times H \times W}. \tag{C.20}$$

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 [42, 43]. 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. Text-based initialization enables intuitively guiding the tracking process through natural language descriptions. This allows high-level user control in specifying objects or attributes of interest. Overall, our text prompt approach unlocks new interface possibilities while capitalizing on diffusion model strengths compared to conventional detector-driven methods.

Pseudo-code for One-shot Training. Alg. C.1 and Alg. C.2 are the pseudo-code for our DIFTracker algorithmic fine-tuning and operating, a proposed approach within the *Tracking-by-Diffusion* paradigm, respectively. The pseudo-code provides a high-level overview of the steps involved in our DIFTracker inplace fine-tuning.

Algorithm C.1 The one-shot fine-tuning pipeline of DIFTracker.

Input: $\mathbf{I}_t$, $\mathbf{I}_{t+1}$, $\mathcal{T} \leftarrow [\tau_\theta(L_t^0) \| \ldots \| \tau_\theta(L_t^{M-1})]$, $T \leftarrow 50$

$\mathbf{z}_0 \leftarrow \mathcal{E}(\mathbf{I}_t)$
$\mathbf{x}_0 \leftarrow \mathcal{E}(\mathbf{I}_{t+1})$
$\mathbf{z}_T \leftarrow inv^{\times T}(\mathbf{z}_0, \mathcal{T})$ *% injected Inversion*
% manipulated Reconstruction
$L_{\text{ELBO}} \leftarrow \text{KL}\big(inv(\mathbf{x}_{T-1}, \mathcal{T}) \| rec(\mathbf{z}_T)\big)$ *%* ℓ_T
for $k \in \{T, \ldots, 2\}$ **do**
    L_{ELBO} += $\text{KL}\big(inv(\mathbf{x}_{k-2}, \mathcal{T}) \| rec(\widehat{\mathbf{z}}_k)\big)$ *%* ℓ_{k-1}
end for
L_{ELBO} −= $\log rec(\widehat{\mathbf{z}}_1)$ *%* ℓ_0
Take gradient descent step on L_{ELBO} to learn $reg(\cdot)$, parameterized by θ

Algorithm C.2 The operating of DIFTracker.

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. C.1
  $\widehat{\mathbf{z}}_T \leftarrow reg(\mathbf{z}_T)$
  for $k \in \{T, \ldots, 1\}$ **do**
    if $k \in [1, T \times 0.8]$ **then**
      $\mathcal{A}_S \mathrel{+}= \sum_{l=1}^{N} Attn_{l,k}(\epsilon_\theta, \epsilon_\theta)$
      $\mathcal{A}_X \mathrel{+}= \sum_{l=1}^{N} Attn_{l,k}(\epsilon_\theta, \tau_\theta)$
    end if
    $\widehat{\mathbf{z}}_k \leftarrow reg(\widehat{\mathbf{z}}_{k+1})$
  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. (14)
  $\mathbf{T} \leftarrow \{L_{t+1}^0, \ldots, L_{t+1}^{M-1}\}$
end for

Noise Resilience. Our DIFTracker is trained to gradually reconstruct the object and entire image context from extreme noise states. This contrasts with other models only seeing clean images throughout training. We hypothesize this diffusion process reduces sensitivity to noise interference. To substantiate, we conduct a noise perturbation analysis under varying JPEG compression levels as in Table C.9. DIFTracker maintains strong performance even with increasing compression ratios that degrade image quality. At a high 70% compression rate, our approach outperforms UNICORN in accuracy. This empirical examination highlights the noise resilience advantage of this diffusion-based method.

Table C.9: JPEG compression analysis for segment tracking performance $\mathcal{J}\&\mathcal{F}$.

Compression Ratio	0%	50%	55%	60%	65%	70%
UNICORN [10]	69.2	$62.5_{(-6.7)}$	$60.8_{(-8.4)}$	$58.5_{(-10.7)}$	$55.8_{(-13.4)}$	$52.4_{(-16.8)}$
DIFTracker	75.7	$75.4_{(-0.3)}$	$74.9_{(-0.8)}$	$73.8_{(-1.9)}$	$70.5_{(-5.2)}$	$65.3_{(-10.4)}$