This ICCV paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Online Dense Point Tracking with Streaming Memory

Qiaole Dong and Yanwei Fu✉
School of Data Science, Fudan University
{qldong18, yanweifu}@fudan.edu.cn

Abstract

*Dense point tracking is a challenging task requiring the continuous tracking of every point in the initial frame throughout a substantial portion of a video, even in the presence of occlusions. Traditional methods use optical flow models to directly estimate long-range motion, but they often suffer from appearance drifting without considering temporal consistency. Recent point tracking algorithms usually depend on sliding windows for indirect information propagation from the first frame to the current one, which is slow and less effective for long-range tracking. To account for temporal consistency and enable efficient information propagation, we present a lightweight and fast model with **S**treaming memory for dense **PO**int **T**racking and online video processing. The **SPOT** framework features three core components: a customized memory reading module for feature enhancement, a sensory memory for short-term motion dynamics modeling, and a visibility-guided splatting module for accurate information propagation. This combination enables SPOT to perform dense point tracking with state-of-the-art accuracy on the CVO benchmark, as well as comparable or superior performance to offline models on sparse tracking benchmarks such as TAP-Vid and RoboTAP. Notably, SPOT with 10× smaller parameter numbers operates at least 2× faster than previous state-of-the-art models while maintaining the best performance on CVO. We will release the models and codes at: https://dqiaole.github.io/SPOT/.*

1. Introduction

A fine-grained and long-term analysis of motion, like what we experience in human vision, has been a long-standing goal in computer vision. This problem has typically been approached through optical flow [17, 30, 47, 50] and point tracking [13, 14, 22, 32]. While each type of motion facilitates various real-world applications [7, 19, 20, 48, 51], neither can fully capture dense, long-range motion in real time and on a frame-by-frame basis for streaming motion perception. Here, we consider dense point tracking, which

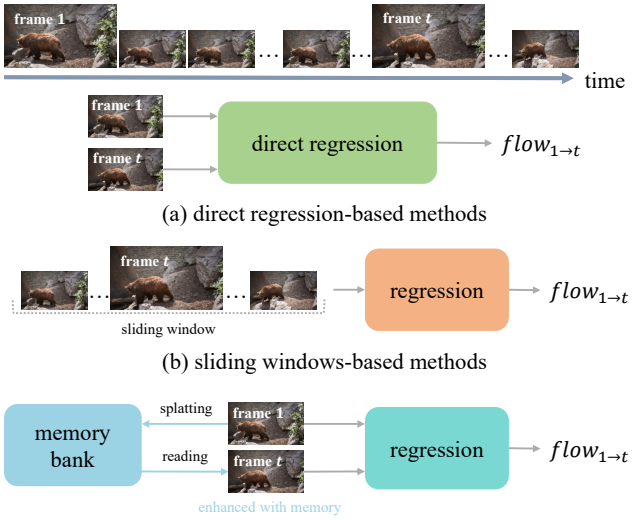


Figure 1. Main frameworks for dense and long-range correspondence estimation. (a) Classical methods employ optical flow models for direct regression of long-range optical flow. (b) Recent sparse point tracking methods rely on sliding windows for information propagation and require offline processing. (c) Our SPOT introduces a fast streaming memory that stores information of first frame and enhances the current frame for better regression.

is essentially the same task as long-range optical flow over many frames, and will use these terms interchangeably.

Classical methods typically utilize the optical flow [17, 30, 47, 50] models, originally trained on adjacent frames, for direct regression of long-range flow as depicted in Fig. 1. Recent work like TAP-Net [13] and MFT [39] has bridged the gap by training on long-range frame pairs and chaining reliable short-range flows respectively. However, they still suffer from appearance drifting and struggle with occlusions without considering the temporal consistency.

As the appearance information of physical points within the first frame needs to be propagated and compared to the one in the target frame, recent point tracking [22, 32, 54] methods typically employ sliding windows with offline processing, *i.e.*, necessitating future frames for improving the accuracy. Despite the super performance of these models, they require a huge amount of time to track every pixel.

Implicit information propagation through sliding windows is also inefficient and less effective for long-range tracking. Recent hybrid method DOT [33] combines optical flow and sparse point tracking for fast and dense point tracking. However, it still inherits the disadvantage of base methods. This raises a question: *Is it possible to achieve both high accuracy and efficiency based on past observation solely?*

We present a novel framework with **S**treaming memory for online dense **PO**int **T**racking (**SPOT**) and real-time video processing. SPOT offers the following advantages:

- *Strong Performance*: On CVO [33, 53], SPOT achieves state-of-the-art performance, outperforming the second one by 0.35 EPE (a 6.8% error reduction) on Extended split. On TAP-Vid [13] and RoboTAP [51], SPOT obtains up to 11.3% relative improvement in AJ from the best online point tracking results.
- *Superior Efficiency*: SPOT tracks 512×512 videos densely at 12.4 FPS on H100, consuming 4.15GB GPU memory. With $10\times$ smaller parameter counts, it operates at least $2\times$ faster than previous state-of-the-art models while still leading the accuracy on CVO [33, 53].
- *Causal Processing*: SPOT processes videos causally, on a frame-by-frame basis without relying on future frames, paving the way toward online dense motion perception.

SPOT consists of three main components: (1) a customized memory reading module based on the assumption of local affinity for feature enhancement; (2) a sensory memory for short-term motion dynamics modeling; (3) a visibility-guided splatting module for accurate long-range information propagation from the first frame. Fig. 1 shows a conceptual comparison with previous frameworks.

The key idea in our approach is to break down the challenging task of long-range information propagation into two simpler steps that can be solved efficiently and accurately. First, we can use splatting [40] to transfer the features from the first frame to the coordinate grid of recent frames. With the accurate long-range optical flow already predicted, splatting ensures the features are mapped to correct positions for the same physical points across frames. Next, there is minimal appearance drifting between the recent frame and the current one, which enables using simple feature similarity (via an attention mechanism) to efficiently retrieve relevant features to enhance the current frame. As a result, regressing with the memory-enhanced feature can give us an accurate long-range optical flow.

The SPOT framework builds on ideas from existing research but introduces several key innovations. First, SPOT uses a memory bank to store distinctive features from the first frame and includes a specialized memory reading module to enhance the feature of current frame for accurate long-range optical flow prediction. In contrast, the recent memory-based model MemFlow [16] stores only short-term motion features and can only handle short-range optical flow between neighboring frames.

Second, SPOT includes an additional sensory memory to model short-term motion dynamics. This sensory memory, updated alternately with the flow, supports accurate long-range flow prediction. In comparison, recent sparse tracking methods [32, 54] handle short-term motion by processing all frames within a sliding window, which is less efficient.

Third, SPOT's regression operator uses a standard optical flow decoder, specifically the RAFT decoder [50]. This approach unifies the architecture for both optical flow and point tracking, meaning any future improvements in optical flow architectures can directly enhance our model.

We conduct experiments on CVO [33, 53], TAP-Vid [13], and RoboTAP [51]. The results demonstrate that SPOT achieves state-of-the-art performance for online tracking across these datasets. Additionally, we validate SPOT's design choices through extensive ablation studies.

2. Related Work

Optical Flow Estimation. Optical flow estimation is traditionally formulated as an energy function minimization problem [2–5, 23, 46, 56], while recent works [16–18, 25, 26, 28, 30, 44, 47, 49, 50, 55, 58] resort to deep learning network trained on large scale dataset for direct generation. However, these works typically focus on optical flow estimation between adjacent frames. Early attempts [11, 12, 29, 38] have tried to chain short-range optical flow and deduce long-range motions. Recently, MFT [39] has improved the pipeline through reliable chaining identification, and AccFlow [53] introduces an occlusion robust backward accumulation strategy. Nonetheless, chaining-based methods have intrinsic limitations, *i.e.*, forward accumulation [39] struggles with occlusion while backward accumulation [53] leads to linear increasing of processing time for a new frame. Our SPOT also builds upon the standard optical flow model [50]. Yet distinct from chaining-based methods, SPOT employs an effective and efficient memory and shows it is indeed possible to achieve both highly accurate results and fast inference speed.

Point Tracking. Point Tracking aims to predict the position and visibility of the same physical point in every other video frame, given the query point in one frame. Since the pioneering works, PIPs [22] and TAP-Net [13], point tracking has attracted a lot of attention and numerous works [14, 32, 33, 36, 51, 54, 57] have emerged. Among them, DOT [33] and Online TAPIR [51] tackle a similar task as ours. Specifically, DOT proposes an optical flow refinement stage for tracking densification. However, it depends on sparse tracks for initialization, which can easily miss small objects. Besides, combining several models also incurs a huge amount of model parameters. As for Online TAPIR, it proposes to replace the temporal convolutions with causal convolutions within TAPIR [14] for

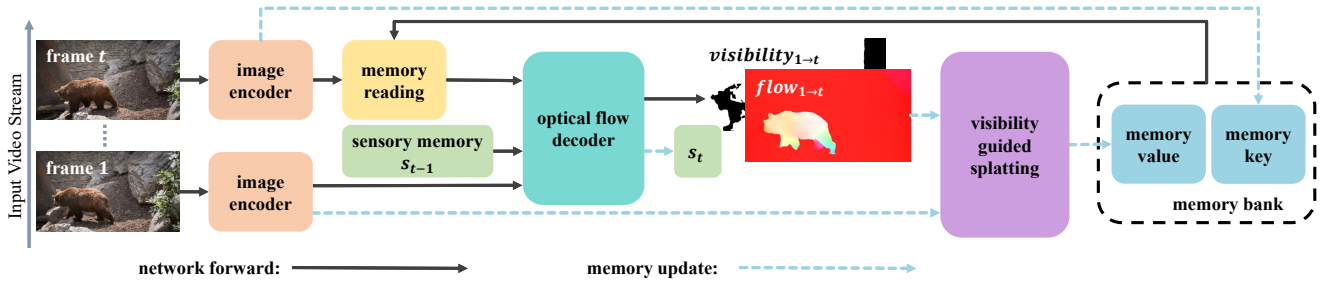


Figure 2. Overview of our SPOT. With streaming video input, we first extract 4× downsampled feature for the current frame. Then the feature is enhanced by cross-attending to the memory to obtain a more discriminative feature. The optical flow and visibility mask are later obtained by a standard optical flow decoder with an additional input as sensory memory that models short-term motion dynamics. Finally, modulating by the visibility mask, we splat the feature of first frame to the memory value with the predicted flow for future usage. The memory key and sensory memory are updated by the current feature and the optical flow decoder, respectively.

online tracking, yet it is infeasible to track pixels densely in a reasonable amount of time. Conversely, SPOT estimates dense point tracking with both high accuracy and efficiency, achieved by a dedicated memory. Besides, leveraging real-world videos for self-supervised finetuning as in BootsTAP [15] is also a promising technique, although it is beyond the scope of our paper.

Memory in Motion Estimation. Memory mechanism has been widely used in various video tasks such as video object segmentation [8–10, 24, 37, 41, 45], video semantic segmentation [42, 52], and video prediction [35]. However, VOS focuses on the identification of single or multiple object-level correspondences. It is non-trivial to directly extend them to dense and pixel-level correspondence tasks. Recently, MemFlow [16] pioneers using memory for optical flow estimation, but can only model short-range motion between adjacent frames. Importantly, our motivation originates from a deep analysis of online dense point tracking, which is naturally suitable for modeling with memory. Furthermore, we propose a novel splatting method for memory updating that is appropriate for long-range propagation.

3. SPOT

3.1. Definition and Overview

Problem Setup. The goal of point tracking is to predict the position $(x, y) \in \mathbb{R}^2$ and visibility $v \in [0, 1]$ of query point $(x^q, y^q) \in \mathbb{R}^2$ in other video frames, which is equivalent to predict the long-range optical flow $\mathbf{f} = (x - x^q, y - y^q)$. In the setting of our online dense point tracking, we have access to all frames before the current timestamp t: $\{\mathbf{I}_1, \mathbf{I}_2, \cdots, \mathbf{I}_t\} \in \mathbb{R}^{t\times H\times W\times 3}$. We aim to predict the optical flow $\mathbf{f}_{1\to t} \in \mathbb{R}^{H\times W\times 2}$ and visibility mask $\mathbf{v}_{1\to t} \in [0, 1]^{H\times W}$ in a streaming fashion.

Overview. Fig. 2 provides an overview of SPOT. First, SPOT encodes the current frame into 4× downsampled feature and conditions it on the memory bank to produce a more discriminative embedding (Sec. 3.2). Then optical flow decoder accepts enhanced feature, reference feature, and an additional sensory memory of shot-range motion dynamics as inputs, and outputs long-range optical flow with visibility mask (Sec. 3.3). The sensory memory is also refreshed with internal motion feature within the flow decoder. SPOT finally updates the memory bank through visibility-guided splatting with predicted flow and reference feature for information propagation (Sec. 3.4). In the following, we will describe each module sequentially and provide more implementation details in Sec. 3.5.

3.2. Feature Enhancement with Memory Reading

Given current video frame $\mathbf{I}_t$, we employ a standard residual convolutional network to extract dense feature at a lower resolution: $\mathbf{F}_t \in \mathbb{R}^{H/4\times W/4\times D}$, where D is the feature dimension. As our memory bank consists of keys and values, the original feature $\mathbf{F}_t$ can serve as the query and the attention mechanism can be used for information retrieval.

Specifically, the original feature $\mathbf{F}_t$ is projected through a linear projection layer for generating the query feature q. Then a dot-product similarity calculation is performed against the memory key followed by a softmax for normalization. This normalized score is used to aggregate the memory value to obtain the readout features $\mathbf{M}_t$. The process can be formulated as:

$$\mathbf{M}_t = \text{Softmax}(1/\sqrt{D_k} \times q \times k^T) \times v, \qquad (1)$$

where $k \in \mathbb{R}^{L\times D_k}$ and $v \in \mathbb{R}^{L\times D_v}$ are keys and values with L elements stored in the memory bank.

Yet, we would like to point out that it is inappropriate to use the readout features $\mathbf{M}_t$ for flow regression directly. As our memory value is obtained through splatting the reference feature $\mathbf{F}_1$ with the previously predicted flow (which will be introduced in Sec. 3.4), there are many blank hole artifacts [1, 31] due to disocclusion [27]. These disocclusion artifacts will hamper the network training and degrade the performance of flow regression. Inspired by the typically used inpainting technique for hole filling [1], we further

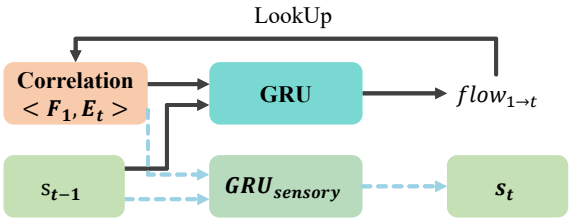


Figure 3. Overview of **optical flow decoder** within Fig. 2. Motion feature is obtained through flow lookup. We update the flow with sensory memory, and update the sensory memory with the final motion feature. For the sake of simplicity, visibility prediction and context feature are omitted here.

propose a fusion layer for artifacts remedy. Fortunately, we find that one single convolution layer performs well enough, and we "inpaint" the feature through the original feature $\mathbf{F}_t$:

$$\mathbf{E}_t = \mathbf{F}_t + \text{Conv}(\mathbf{F}_t \oplus \mathbf{M}_t), \tag{2}$$

where $\oplus$ denotes concatenation and $\mathbf{E}_t$ is the final enhanced feature that is ready for the next stage of flow regression.

3.3. Flow Decoding and Sensory Memory

With the memory-enhanced feature $\mathbf{E}_t$ and the reference feature $\mathbf{F}_1$, we first compute the 4D correlation volume between all pairs following optical flow literature [50]. The motion feature f_m^i is obtained through a lightweight motion encoder, taking current estimation $\mathbf{f}_{1\to t}^i$, $\mathbf{v}_{1\to t}^i$, and correlation volume as input. Finally, we produce an update direction for flow and visibility with a GRU unit as in [33]:

$$\Delta f_{1\to t}^i, \Delta v_{1\to t}^i, h_t^i = \text{GRU}(h_t^{i-1}, f_c, f_m^i, s_{t-1}), \tag{3}$$

where h_t^i is the hidden state of GRU, f_c is the context feature of $\mathbf{I}_1$, and $s_{t-1} \in \mathbb{R}^{H/4\times W/4\times D_s}$ is the sensory memory of motion history. We perform a total of N refinement steps of GRU and get the final optical flow, visibility mask, and corresponding motion feature f_m^N. Importantly, we update the sensory memory with final motion feature by an additional GRU for short-term motion dynamic modeling:

$$s_t = \text{GRU}_{sensory}(s_{t-1}, f_m^N). \tag{4}$$

An overview of this process is illustrated in Fig. 3.

3.4. Visibility-guided Splatting

The key to our formulation is to obtain discriminative features through memory reading. Recall that SPOT relies on feature similarity to aggregate memory value, while similarity score is only accurate within a handful of frames before the current frame due to appearance drifting. Therefore, we need to continually propagate the discriminative features of reference frame to latest frame. Fortunately, the predicted optical flow $\mathbf{f}_{1\to t}^N$ can be used for such long-range information propagation with both high accuracy and efficiency.

To achieve this, forward warping via splatting [40] becomes a natural choice. Specifically, for every position (x_1, y_1), we calculate its position in the target frame by adding the optical flow. Then bilinear kernel $b(\cdot)$ (details in Supp.) is employed for weighted summation as follows:

$$F_t^{\Sigma}\left[(x_t, y_t)\right] = \sum_{(x_1,y_1)} b(\Delta)\cdot \mathbf{F}_1\left[(x_1, y_1)\right], \tag{5}$$

$$\text{where} \quad \Delta = (x_1, y_1) + \mathbf{f}_{1\to t}^N\left[(x_1, y_1)\right] - (x_t, y_t). \tag{6}$$

We denote the splatting results as $\overrightarrow{\sum}(\mathbf{F}_1, \mathbf{f}_{1\to t}) = F_t^{\Sigma}$. However, this splatting will lead to inconsistencies artifacts [40] for occlusion regions. Therefore, we resort to our predicted visibility mask for occlusion handling and get the final splatting results as follows:

$$F_t^{1\to t} = \frac{\overrightarrow{\sum}(\mathbf{v}_{1\to t}\cdot \mathbf{F}_1, \mathbf{f}_{1\to t})}{\overrightarrow{\sum}(\mathbf{v}_{1\to t}, \mathbf{f}_{1\to t})}, \tag{7}$$

where we mask the invisible regions and normalize the results with visibility score.

Lastly, the memory bank maintains two FIFO queues of keys and values, caching a window of L recent first-frame-derived features. We append the splatting results $F_t^{1\to t}$ into the queue of memory value. As for the memory key, we reuse the query feature q of $\mathbf{F}_t$ for simplicity.

3.5. Implementation Details

Here, we describe more implementation details of SPOT.
Warm-Start. As SPOT is applied for streaming videos, we employ previous information to warm-start the estimation of next frame. Particularly, we initialize the hidden state h_t^0 of GRU with h_{t-1}^N and initialize the flow $\mathbf{f}_{1\to t}^0$ by one step extrapolation: $\mathbf{f}_{1\to t}^0 = \mathbf{f}_{1\to t-1}^0 + 2\times(\mathbf{f}_{1\to t-1}^N - \mathbf{f}_{1\to t-1}^0)$.
Training. The loss function of SPOT consists of the l_1 loss [50] for the predicted flows and binary cross-entropy loss for the visibility prediction. SPOT is first trained on synthetic optical flow dataset, *i.e.*, Kubric-CVO [53], for 500k steps at resolution 384×384. Then it is finetuned on Kubric-MOVi-F [21] for 100k steps with 24-frame videos of resolution 384×384. During training, we set the iteration of flow decoding N to 12. If not otherwise mentioned, we set N to 16 during inference. To reduce the drifting effect, we set the memory bank length L to 3.

4. Experiments

We conduct evaluations on three benchmarks: long-range optical flow benchmark CVO [33, 53], sparse point tracking benchmark TAP-Vid [13], and RoboTAP [51]. SPOT is mainly compared with recent online methods, *e.g.* MFT [39] and Online TAPIR [51]. We also consider several offline methods for comparison, namely CoTracker2 [32],

Table 1. Long-range optical flow estimation on CVO. We evaluate dense optical flow predictions from the first to the last frame of videos. DOT [33] here uses CoTracker2 [32] for the first stage of sparse point tracking. * uses the flow of neighboring frames as an initialization for more distant frames. DOT† represents evaluated the first stage of DOT [33] in an online fashion with sliding windows of 2.

Method	CVO (Clean)		CVO (Final)		CVO (Extended)	
	EPE ↓ (all / vis / occ)	OA ↑	EPE ↓ (all / vis / occ)	OA ↑	EPE ↓ (all / vis / occ)	OA ↑
Offline						
PIPs++ [57]	9.05 / 6.62 / 21,5	-	9.49 / 7.06 / 22.0	-	18.4 / 10.0 / 32.1	-
TAPIR [14]	4.15 / 1.48 / 16.5	96.9	4.59 / 1.88 / 17.2	96.6	22.6 / 4.70 / 49.7	88.3
CoTracker2 [32]	1.50 / 0.90 / 4.41	97.1	1.47 / 0.94 / 4.20	97.0	5.45 / 4.03 / 8.30	88.2
SpatialTracker [54]	1.84 / 1.32 / 4.72	-	1.88 / 1.37 / 4.68	-	5.53 / 4.18 / 8.68	-
DOT [33]	**1.34 / 0.76 / 4.11**	**97.6**	**1.37 / 0.84 / 4.06**	**97.5**	**5.12 / 3.65 / 7.61**	**88.9**
Online						
RAFT [50]	2.82 / 1.70 / 8.01	90.2	2.88 / 1.79 / 7.89	89.7	28.6 / 21.6 / 41.0	62.3
GMA [30]	2.90 / 1.91 / 7.63	91.0	2.92 / 1.89 / 7.48	90.6	30.0 / 22.8 / 42.6	62.0
RAFT* [50]	2.19 / 1.20 / 6.77	90.1	2.35 / 1.35 / 6.76	89.6	21.1 / 23.0 / 25.6	63.6
GMA* [30]	2.16 / 1.19 / 6.59	91.1	2.25 / 1.28 / 6.53	90.8	21.0 / 22.4 / 25.2	64.0
MFT [39]	2.91 / 1.39 / 9.93	95.8	3.16 / 1.56 / 10.3	95.7	21.4 / 9.20 / 41.8	85.5
AccFlow [53]	1.69 / 1.08 / 4.70	84.8	1.73 / 1.15 / 4.63	84.4	36.7 / 28.1 / 52.9	58.8
DOT† [33]	1.92 / 1.07 / 6.10	**97.1**	1.98 / 1.17 / 6.02	**97.1**	12.1 / 7.06 / 20.7	87.0
SPOT (Ours)	**1.11 / 0.60 / 3.61**	96.8	**1.23 / 0.72 / 3.67**	96.5	**4.77 / 3.58 / 6.96**	**88.6**

TAPTR [36], and DOT [33]. We further provide a more comparable version of DOT by reducing their sliding window size to 2 (DOT†) for an online dense point tracking evaluation, which is the most similar protocol with reasonable inference speed we pursue here. We present the evaluation protocols and comparison results as follows.

4.1. CVO

CVO test set [53] originally includes Clean and Final pass, each containing 536 videos of 7 frames at 60 FPS. Lately, DOT [33] introduces the Extended set with another 500 videos of 48 frames at 24 FPS. All videos are provided with groundtruth optical flow and visibility mask between first and last frame. We utilize end-point-error (EPE) within all, visible, and occluded regions for flow evaluation, and occlusion accuracy (OA) for visibility prediction evaluation.

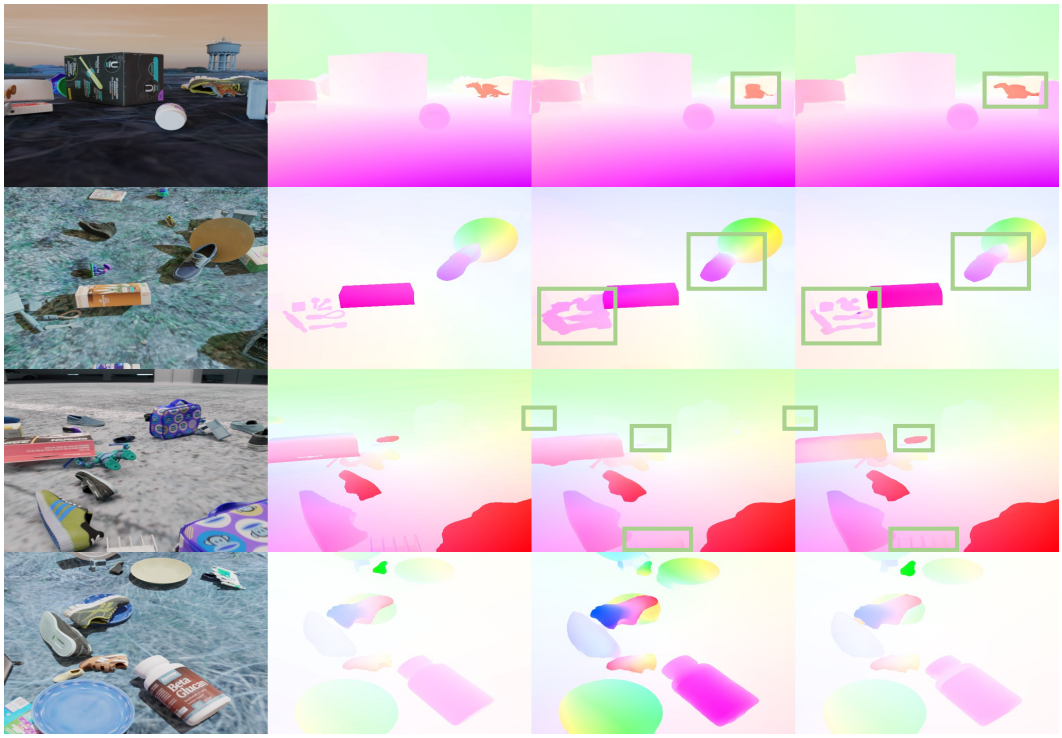
As shown in Tab. 1, SPOT achieves state-of-the-art performance on all three CVO test sets in terms of EPE. Particularly, SPOT achieves a 60.5% reduction from 12.08 to 4.77 compared to the online DOT† on the Extended set. It even outperforms all offline methods across three sets for flow prediction. This is due to CVO only evaluating the flow between the first and last frame, there is no future frame that can be utilized by offline methods to improve the performance. This demonstrates that SPOT enables the exploitation of past information more effectively. Besides, SPOT outperforms MFT in terms of occlusion accuracy and performs comparably with DOT†. Fig. 4 provides a qualitative comparison with DOT, SPOT successfully predicts the flow for small objects and suffers less drifting than DOT.

4.2. TAP-Vid and RoboTAP

TAP-Vid [13] provides 3 datasets with different scenarios: 30 real-world videos from DAVIS [43], over 1,000 human actions videos from Kinetics [6], and 50 synthetic videos for robotic manipulation from RGB-Stacking [34]. In addition, RoboTAP [51] further provides a dataset with 265 real-world videos of robotic manipulation tasks. We evaluate SPOT on DAVIS in both "first" and "strided" query modes [13] while evaluating other datasets in "first" mode solely. Following the standard evaluation protocol of TAP-Vid, we first downsample the videos to 256×256 and report occlusion accuracy (OA), position accuracy ($< \delta^x_{avg}$) for visible points, and average jaccard (AJ) which evaluates both occlusion and position accuracy.

The quantitative comparisons are reported in Tab. 2, which shows SPOT consistently outperforms other online methods on all datasets. Specifically, when compared to previous state-of-the-art online tracking models, the relative improvements of SPOT up to 11.3% in average jaccard (AJ), 9.9% in position accuracy ($< \delta^x_{avg}$), and 4.1% in occlusion accuracy (OA). SPOT even outperforms some recent offline tracking methods slightly, *e.g.*, beats CoTracker2 [32] on all datasets and performs better than SpatialTracker [54] in terms of AJ. Furthermore, SPOT achieves the best AJ in Kinetics.

We further provide qualitative comparisons with Online TAPIR [51], which is currently the best method for online tracking, on challenging videos from DAVIS in Fig. 5. SPOT outperforms Online TAPIR in handling complex occlusion, estimating better and consistent motion state of the car and people behind distractors (the first and sec-



(a) Reference Frame (b) Ground Truth (c) DOT (d) **SPOT**

Figure 4. Qualitative comparison on the CVO (Extended). Notable areas are marked by a bounding box. Please zoom in for details.

Table 2. Point tracking results on TAP-Vid and RoboTAP. DOT† represents evaluated in an online fashion with sliding windows of 2.

Method	DAVIS (First)			DAVIS (Strided)			RGB-S. (First)			RoboTAP (First)			Kinetics (First)		
	AJ ↑	$< \delta^x_{avg}$ ↑	OA ↑	AJ ↑	$< \delta^x_{avg}$ ↑	OA ↑	AJ ↑	$< \delta^x_{avg}$ ↑	OA ↑	AJ ↑	$< \delta^x_{avg}$ ↑	OA ↑	AJ ↑	$< \delta^x_{avg}$ ↑	OA ↑
Offline															
PIPs [22]	42.2	64.8	77.7	52.4	70.0	83.6	-	-	-	29.5	-	-	31.7	53.7	72.9
TAPIR [14]	56.2	70.0	86.5	61.3	73.6	88.8	55.5	69.7	88.0	59.6	73.4	87.0	49.6	64.2	85.0
CoTracker2 [32]	60.8	74.8	88.4	65.9	79.4	89.9	60.5	73.3	83.5	58.6	70.6	87.0	48.4	62.2	83.2
SpatialTracker [54]	61.1	**76.3**	89.5	-	-	-	63.5	77.6	88.2	-	-	-	**50.1**	**65.9**	**86.9**
TAPTR [36]	**63.0**	76.1	**91.1**	66.3	79.2	91.0	60.8	76.2	87.0	60.1	**75.3**	86.9	49.0	64.4	85.2
DOT [33]	62.2	76.1	89.9	**67.8**	**80.6**	**91.1**	**77.5**	**87.4**	**92.6**	**61.5**	73.7	**87.9**	48.4	63.8	85.2
Online															
TAP-Net [13]	33.0	48.6	78.8	38.4	53.1	82.3	53.5	68.1	86.3	45.1	-	-	38.5	54.4	80.6
MFT [39]	47.3	66.8	77.8	56.1	70.8	86.9	-	-	-	-	-	-	39.6	60.4	72.7
DOT† [33]	53.3	67.8	85.4	60.1	73.8	87.3	61.3	75.3	86.5	51.9	62.9	79.9	45.3	58.0	81.4
Online TAPIR [51]	56.2	69.3	84.6	58.3	70.6	84.6	65.9	78.6	89.5	57.9	69.9	86.1	49.6	61.1	83.1
SPOT (Ours)	**61.5**	**75.0**	**88.9**	**66.9**	**79.4**	**90.2**	**73.3**	**86.4**	**90.3**	**60.7**	**73.6**	**87.5**	**50.2**	**62.7**	**85.5**

ond examples of Fig. 5), achieves better tracking during multi-subjects interaction (the third example), and effectively tracks small part of the animal (the fourth example). Animated video results can be found in our project page.

4.3. Timing and Parameter Counts

We present inference time and parameter counts in Fig. 6. Accuracy is determined by performance on the CVO (Final) set. The parameter number of SPOT is only 8.7M, which is an order of magnitude smaller than the number of parameters of other point tracking methods, *e.g.*, 29.3M for Online TAPIR and 56.5M for DOT. For video input of 512×512, SPOT can track all points within the first frame and runs at 12.4 FPS on one H100 GPU, which is much faster than sparse tracking methods like Online TAPIR and CoTracker2. And SPOT also runs faster than DOT and DOT† in our hardware without sacrificing performance. Be-

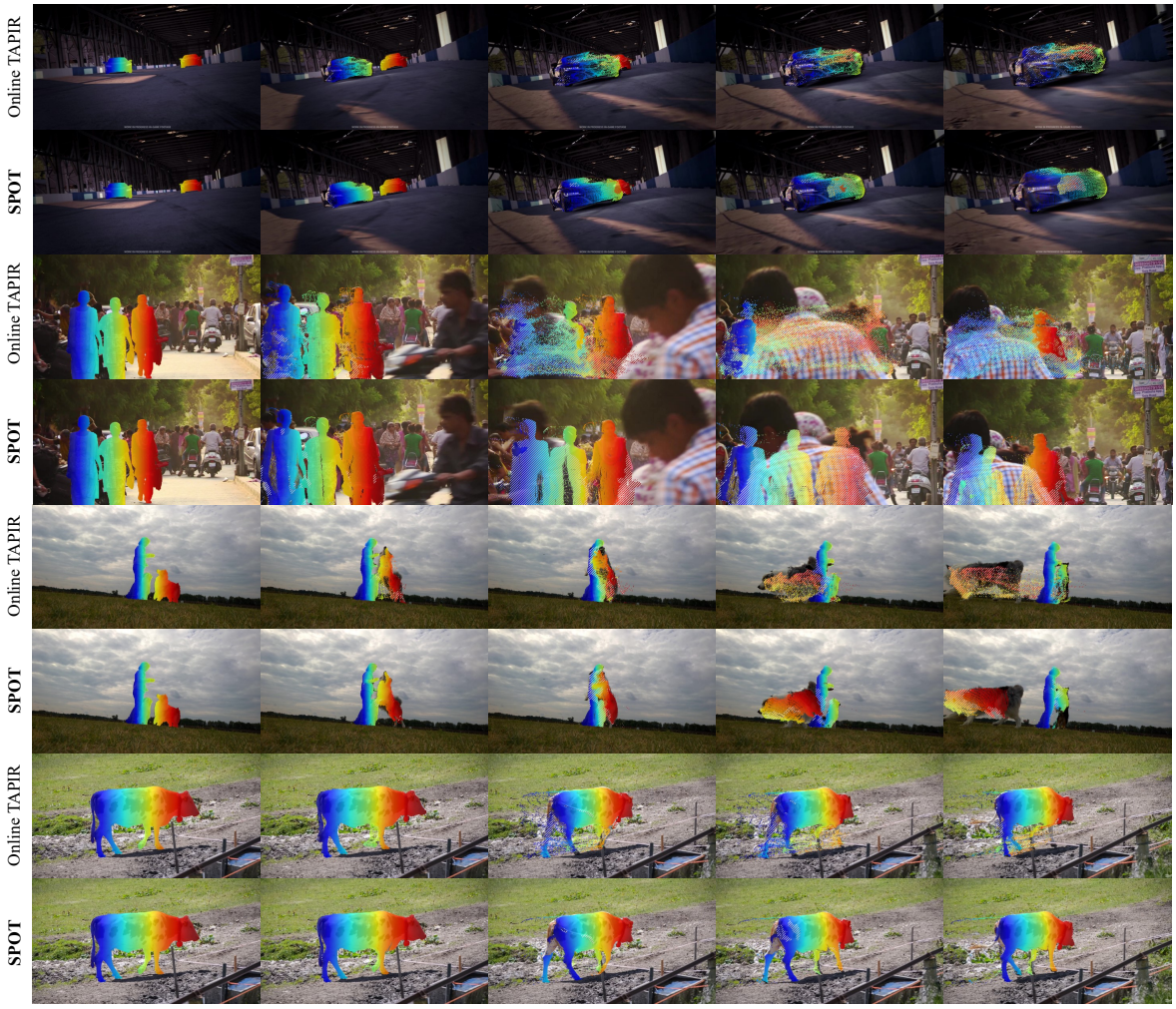


Figure 5. Qualitative comparison on DAVIS. For each sequence, we show tracking results of Online TAPIR [51] and SPOT. Only foreground points of the first frame are visualized, each point is displayed with a different color and overlayed with white stripes if occluded.

sides, our parameter number and inference speed are similar to recent optical flow models while performing much better than them in terms of dense point tracking.

4.4. Ablation study

In this section, we perform a series of ablation studies on SPOT to show the relative importance of each module. All experiments are conducted on CVO (Extended). For time-saving, SPOT is trained on 10-frame videos in Tab. 3.

Module Ablation. We first experiment with the importance of each network module as in Tab. 3. Specifically, we first remove the feature fusion module (Eq. (2)) during the stage of feature enhancement. Without feature fusion, the blank hole artifacts will soon take over the network and easily lead to NaN during training. Furthermore, removing the memory bank and memory reading together results in a classical optical flow estimation model. Though trained on point tracking dataset Kubric-MOVi-F [21], the model without memory bank performs much worse than the full model, which confirms the contribution of memory bank. Also, the results benefit from the sensory memory, which makes the model aware of short-term motion dynamics and improves the EPE from 8.64 to 6.42. Besides, an additional query projector for memory reading can improve the results slightly.

Splatting Type. We then present the ablation of different splatting strategies as in the second part of Tab. 3. The splatting strategy used in Eq. (7) is dubbed as linear splatting in [40]. Average splatting uses all one tensor for mask and normalization, while summation splatting corresponds to Eq. (5). The softmax splatting proposed in [40] uses additional exponential functions for mask calculation. As shown in Tab. 3, the linear splatting achieves the best performance, outperforming the second one, softmax, by 0.62 EPE.

Warm-Start. We finally validate the effectiveness of warm-start strategy in the last part of Tab. 3. Initializing the hidden state h_t^0 of GRU with h_{t-1}^N indeed results in significant per-

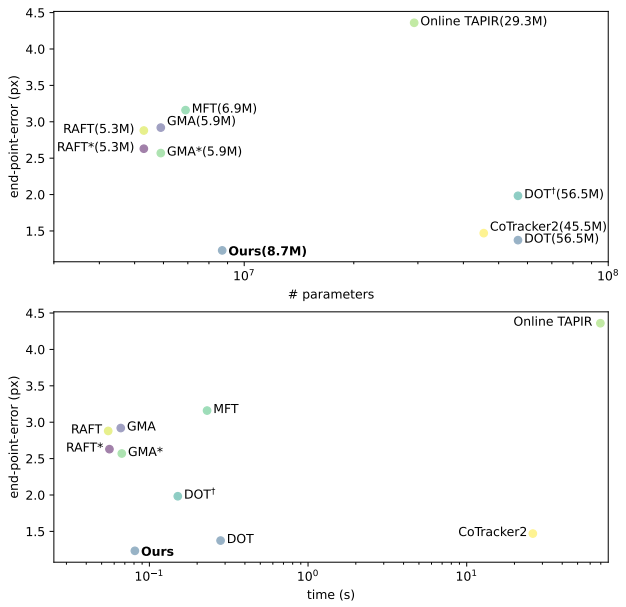


Figure 6. Plots comparing parameter counts (M) and inference time (s) vs. accuracy. Accuracy is measured by the EPE on the CVO (Final) test set. *Top*: Parameter count vs. accuracy compared to other methods. Our method is more parameter efficient while achieving lower EPE than previous state-of-the-art methods. *Bottom*: Inference time per frame of 512×512 resolution vs. accuracy on one NVIDIA H100 GPU.

Table 3. Ablation studies on CVO (Extended). Settings used in our final model are underlined. We use 10-frame videos for training.

Experiment	Method	EPE ↓ (all / vis / occ)	OA ↑
Module Ablation	Full	**6.42 / 3.86 / 9.98**	**88.5**
	- Feature fusion	NaN	NaN
	- Memory bank	38.98 / 28.34 / 57.15	78.8
	- Sensory memory	8.64 / 4.55 / 14.60	88.2
	- Query projector	6.48 / 3.88 / 9.82	88.3
Splatting Type	Linear	**6.42 / 3.86 / 9.98**	**88.5**
	Average	7.34 / 3.94 / 11.41	88.5
	Softmax	7.04 / 4.35 / 10.60	88.5
	Summation	7.17 / 3.96 / 11.17	88.5
Warm-Start	Full	**6.42 / 3.86 / 9.98**	**88.5**
	- Hidden state	7.94 / 3.81 / 13.17	88.3
	- Flow	6.49 / 3.92 / 10.05	88.4

formance gains, improving the EPE by 23.7%. This may be because the hidden state also carries the motion information from the previous frame. Besides, warm-start the flow also improves the performance slightly.

Training Video Length. We finally probe the importance of training video length to the final performance, as demonstrated in Tab. 4. We experiment with video length (7, 10, and 24) during training and report the performance on CVO (Extended). Generally, we find that increasing the video length of training can improve the model performance. However, it also brings a huge memory consumption burden to our GPU hardware during training. Therefore, we finally choose to use 24-frame videos for training, which just fit into a NVIDIA 80GB H100 GPU.

Table 4. Ablations of video/memory length on CVO (Extended).

Experiment	Length	EPE ↓ (all / vis / occ)	OA ↑
Video Length	7	7.14 / 3.94 / 11.56	88.5
	10	6.42 / 3.86 / 9.98	88.5
	24	**4.77 / 3.58 / 6.96**	**88.6**
Memory Length	1	5.00 / 3.75 / 7.18	88.2
	3	**4.77 / 3.58 / 6.96**	**88.6**
	6	5.06 / 3.73 / 7.37	88.0

Discussion. We have demonstrated empirically that online dense point tracking can be resolved effectively by the memory mechanism. The intuition is that discriminative information can be propagated from the first frame to the latest frame within memory via already predicted accurate optical flow effectively, and this information can be readout by the attention mechanism based on feature similarity. As there is typically one or two timesteps between the current frame and the frame within memory, the feature similarity is accurate enough for information retrieval. This two-step information propagation mechanism is the cornerstone of the success of our approach.

Limitation. However, the success of two-step propagation depends on the success of each step. Therefore, for accurate attention-based readout without suffering from the drifting problem, the length of the memory bank is typically small (3 in SPOT as verified in Tab. 4). If there is a video with a extreme long time of occlusion, then SPOT may degrades to pairwise method, which is the main limitation of SPOT. To deal with such scenarios, one possible future work is to introduce persistent object-level memory, for the objects within the first frame, that updates adaptively and survives through the long occlusion. Besides, our SPOT is also not robust to extreme fast motion and texture-less region. And we provide more failure cases in Supplementary.

5. Conclusion

We introduced SPOT, a new framework designed for efficient and accurate online dense point tracking. At the core of SPOT is a memory-based approach that enables effective information propagation across frames, supported by our feature enhancement and visibility-guided splatting modules. Additionally, SPOT's sensory memory efficiently models short-term motion dynamics. SPOT achieves state-of-the-art results on CVO, outperforming other online methods and matching the accuracy of recent offline models on TAP-Vid and RoboTAP. With its low parameter count and fast processing speed, SPOT brings us closer to real-time, frame-by-frame motion tracking, opening up new possibilities for practical applications.

Acknowledgements

This work was sponsored by Doubao Fund, ByteDance. Yanwei Fu is also with Shanghai Innovation Institute, Institute of Trustworthy Embodied AI, Fudan University, and ISTBI–ZJNU Algorithm Centre for Brain-inspired Intelligence, Zhejiang Normal University.

References

[1] Filippo Aleotti, Matteo Poggi, and Stefano Mattoccia. Learning optical flow from still images. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15196–15206, 2021. 3

[2] Michael J Black and Padmanabhan Anandan. A framework for the robust estimation of optical flow. In *1993 (4th) International Conference on Computer Vision*, pages 231–236. IEEE, 1993. 2

[3] Michael J Black and Paul Anandan. The robust estimation of multiple motions: Parametric and piecewise-smooth flow fields. *Computer vision and image understanding*, 63(1):75–104, 1996.

[4] Thomas Brox, Andrés Bruhn, Nils Papenberg, and Joachim Weickert. High accuracy optical flow estimation based on a theory for warping. In *European conference on computer vision*, pages 25–36. Springer, 2004.

[5] Andrés Bruhn, Joachim Weickert, and Christoph Schnörr. Lucas/kanade meets horn/schunck: Combining local and global optic flow methods. *International journal of computer vision*, 61(3):211–231, 2005. 2

[6] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308, 2017. 5

[7] Weirong Chen, Le Chen, Rui Wang, and Marc Pollefeys. Leap-vo: Long-term effective any point tracking for visual odometry. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19844–19853, 2024. 1

[8] Ho Kei Cheng and Alexander G Schwing. Xmem: Long-term video object segmentation with an atkinson-shiffrin memory model. In *European Conference on Computer Vision*, pages 640–658. Springer, 2022. 3

[9] Ho Kei Cheng, Yu-Wing Tai, and Chi-Keung Tang. Rethinking space-time networks with improved memory coverage for efficient video object segmentation. *Advances in Neural Information Processing Systems*, 34:11781–11794, 2021.

[10] Ho Kei Cheng, Seoung Wug Oh, Brian Price, Joon-Young Lee, and Alexander Schwing. Putting the object back into video object segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3151–3161, 2024. 3

[11] Tomas Crivelli, Pierre-Henri Conze, Philippe Robert, and Patrick Pérez. From optical flow to dense long term correspondences. In *2012 19th IEEE International Conference on Image Processing*, pages 61–64. IEEE, 2012. 2

[12] Tomas Crivelli, Matthieu Fradet, Pierre-Henri Conze, Philippe Robert, and Patrick Pérez. Robust optical flow integration. *IEEE Transactions on Image Processing*, 24(1): 484–498, 2014. 2

[13] Carl Doersch, Ankush Gupta, Larisa Markeeva, Adria Recasens, Lucas Smaira, Yusuf Aytar, Joao Carreira, Andrew Zisserman, and Yi Yang. Tap-vid: A benchmark for tracking any point in a video. *Advances in Neural Information Processing Systems*, 35:13610–13626, 2022. 1, 2, 4, 5, 6

[14] Carl Doersch, Yi Yang, Mel Vecerik, Dilara Gokay, Ankush Gupta, Yusuf Aytar, Joao Carreira, and Andrew Zisserman. Tapir: Tracking any point with per-frame initialization and temporal refinement. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10061–10072, 2023. 1, 2, 5, 6

[15] Carl Doersch, Pauline Luc, Yi Yang, Dilara Gokay, Skanda Koppula, Ankush Gupta, Joseph Heyward, Ignacio Rocco, Ross Goroshin, João Carreira, et al. Bootstap: Bootstrapped training for tracking-any-point. In *Proceedings of the Asian Conference on Computer Vision*, pages 3257–3274, 2024. 3

[16] Qiaole Dong and Yanwei Fu. Memflow: Optical flow estimation and prediction with memory. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19068–19078, 2024. 2, 3

[17] Qiaole Dong, Chenjie Cao, and Yanwei Fu. Rethinking optical flow from geometric matching consistent perspective. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1337–1347, 2023. 1

[18] Alexey Dosovitskiy, Philipp Fischer, Eddy Ilg, Philip Hausser, Caner Hazirbas, Vladimir Golkov, Patrick Van Der Smagt, Daniel Cremers, and Thomas Brox. Flownet: Learning optical flow with convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2758–2766, 2015. 2

[19] Chen Gao, Ayush Saraf, Jia-Bin Huang, and Johannes Kopf. Flow-edge guided video completion. In *European Conference on Computer Vision*, pages 713–729. Springer, 2020. 1

[20] Daniel Geng, Max Hamilton, and Andrew Owens. Comparing correspondences: Video prediction with correspondence-wise losses. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3365–3376, 2022. 1

[21] Klaus Greff, Francois Belletti, Lucas Beyer, Carl Doersch, Yilun Du, Daniel Duckworth, David J Fleet, Dan Gnanapragasam, Florian Golemo, Charles Herrmann, et al. Kubric: A scalable dataset generator. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3749–3761, 2022. 4, 7

[22] Adam W Harley, Zhaoyuan Fang, and Katerina Fragkiadaki. Particle video revisited: Tracking through occlusions using point trajectories. In *European Conference on Computer Vision*, pages 59–75. Springer, 2022. 1, 2, 6

[23] Berthold KP Horn and Brian G Schunck. Determining optical flow. *Artificial intelligence*, 17(1-3):185–203, 1981. 2

[24] Li Hu, Peng Zhang, Bang Zhang, Pan Pan, Yinghui Xu, and Rong Jin. Learning position and target consistency for memory-based video object segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4144–4154, 2021. 3

[25] Zhaoyang Huang, Xiaoyu Shi, Chao Zhang, Qiang Wang, Ka Chun Cheung, Hongwei Qin, Jifeng Dai, and Hongsheng Li. Flowformer: A transformer architecture for optical flow. *arXiv preprint arXiv:2203.16194*, 2022. 2

[26] Tak-Wai Hui, Xiaoou Tang, and Chen Change Loy. Liteflownet: A lightweight convolutional neural network for optical flow estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8981–8989, 2018. 2

[27] Eddy Ilg. *Estimating optical flow with convolutional neural networks*. PhD thesis, Dissertation, Universität Freiburg, 2019, 2020. 3

[28] Eddy Ilg, Nikolaus Mayer, Tonmoy Saikia, Margret Keuper, Alexey Dosovitskiy, and Thomas Brox. Flownet 2.0: Evolution of optical flow estimation with deep networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2462–2470, 2017. 2

[29] Joel Janai, Fatma Guney, Jonas Wulff, Michael J Black, and Andreas Geiger. Slow flow: Exploiting high-speed cameras for accurate and diverse optical flow reference data. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3597–3607, 2017. 2

[30] Shihao Jiang, Dylan Campbell, Yao Lu, Hongdong Li, and Richard Hartley. Learning to estimate hidden motions with global motion aggregation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9772–9781, 2021. 1, 2, 5

[31] Oğuzhan Fatih Kar, Teresa Yeo, Andrei Atanov, and Amir Zamir. 3d common corruptions and data augmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18963–18974, 2022. 3

[32] Nikita Karaev, Ignacio Rocco, Benjamin Graham, Natalia Neverova, Andrea Vedaldi, and Christian Rupprecht. Cotracker: It is better to track together. *arXiv preprint arXiv:2307.07635*, 2023. 1, 2, 4, 5, 6

[33] Guillaume Le Moing, Jean Ponce, and Cordelia Schmid. Dense optical tracking: connecting the dots. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19187–19197, 2024. 2, 4, 5, 6

[34] Alex X Lee, Coline Manon Devin, Yuxiang Zhou, Thomas Lampe, Konstantinos Bousmalis, Jost Tobias Springenberg, Arunkumar Byravan, Abbas Abdolmaleki, Nimrod Gileadi, David Khosid, et al. Beyond pick-and-place: Tackling robotic stacking of diverse shapes. In *5th Annual Conference on Robot Learning*, 2021. 5

[35] Sangmin Lee, Hak Gu Kim, Dae Hwi Choi, Hyung-Il Kim, and Yong Man Ro. Video prediction recalling long-term motion context via memory alignment learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3054–3063, 2021. 3

[36] Hongyang Li, Hao Zhang, Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, and Lei Zhang. Taptr: Tracking any point with transformers as detection. In *European Conference on Computer Vision*, pages 57–75. Springer, 2025. 2, 5, 6

[37] Mingxing Li, Li Hu, Zhiwei Xiong, Bang Zhang, Pan Pan, and Dong Liu. Recurrent dynamic embedding for video object segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1332–1341, 2022. 3

[38] SukHwan Lim, John G Apostolopoulos, and AE Gamal. Optical flow estimation using temporally oversampled video. *IEEE Transactions on Image Processing*, 14(8):1074–1087, 2005. 2

[39] Michal Neoral, Jonáš Šerỳch, and Jiří Matas. Mft: Long-term tracking of every pixel. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 6837–6847, 2024. 1, 2, 4, 5, 6

[40] Simon Niklaus and Feng Liu. Softmax splatting for video frame interpolation. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2020. 2, 4, 7

[41] Seoung Wug Oh, Joon-Young Lee, Ning Xu, and Seon Joo Kim. Video object segmentation using space-time memory networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9226–9235, 2019. 3

[42] Matthieu Paul, Martin Danelljan, Luc Van Gool, and Radu Timofte. Local memory attention for fast video semantic segmentation. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1102–1109. IEEE, 2021. 3

[43] Federico Perazzi, Jordi Pont-Tuset, Brian McWilliams, Luc Van Gool, Markus Gross, and Alexander Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 724–732, 2016. 5

[44] Anurag Ranjan and Michael J Black. Optical flow estimation using a spatial pyramid network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4161–4170, 2017. 2

[45] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024. 3

[46] Jerome Revaud, Philippe Weinzaepfel, Zaid Harchaoui, and Cordelia Schmid. Epicflow: Edge-preserving interpolation of correspondences for optical flow. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1164–1172, 2015. 2

[47] Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan Kautz. Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8934–8943, 2018. 1, 2

[48] Shuyang Sun, Zhanghui Kuang, Lu Sheng, Wanli Ouyang, and Wei Zhang. Optical flow guided feature: A fast and robust motion representation for video action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1390–1399, 2018. 1

[49] Shangkun Sun, Yuanqi Chen, Yu Zhu, Guodong Guo, and Ge Li. Skflow: Learning optical flow with super kernels. *arXiv preprint arXiv:2205.14623*, 2022. 2

[50] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *European conference on computer vision*, pages 402–419. Springer, 2020. 1, 2, 4, 5

[51] Mel Vecerik, Carl Doersch, Yi Yang, Todor Davchev, Yusuf Aytar, Guangyao Zhou, Raia Hadsell, Lourdes Agapito, and Jon Scholz. Robotap: Tracking arbitrary points for few-shot visual imitation. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5397–5403. IEEE, 2024. 1, 2, 4, 5, 6, 7

[52] Hao Wang, Weining Wang, and Jing Liu. Temporal memory attention for video semantic segmentation. In *2021 IEEE International Conference on Image Processing (ICIP)*, pages 2254–2258. IEEE, 2021. 3

[53] Guangyang Wu, Xiaohong Liu, Kunming Luo, Xi Liu, Qingqing Zheng, Shuaicheng Liu, Xinyang Jiang, Guangtao Zhai, and Wenyi Wang. Accflow: Backward accumulation for long-range optical flow. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12119–12128, 2023. 2, 4, 5

[54] Yuxi Xiao, Qianqian Wang, Shangzhan Zhang, Nan Xue, Sida Peng, Yujun Shen, and Xiaowei Zhou. Spatialtracker: Tracking any 2d pixels in 3d space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20406–20417, 2024. 1, 2, 5, 6

[55] Haofei Xu, Jing Zhang, Jianfei Cai, Hamid Rezatofighi, and Dacheng Tao. Gmflow: Learning optical flow via global matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8121–8130, 2022. 2

[56] Christopher Zach, Thomas Pock, and Horst Bischof. A duality based approach for realtime tv-l 1 optical flow. In *Joint pattern recognition symposium*, pages 214–223. Springer, 2007. 2

[57] Yang Zheng, Adam W Harley, Bokui Shen, Gordon Wetzstein, and Leonidas J Guibas. Pointodyssey: A large-scale synthetic dataset for long-term point tracking. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19855–19865, 2023. 2, 5

[58] Zihua Zheng, Ni Nie, Zhi Ling, Pengfei Xiong, Jiangyu Liu, Hao Wang, and Jiankun Li. Dip: Deep inverse patchmatch for high-resolution optical flow. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8925–8934, 2022. 2