

S. Guttikonda\*, M. Neidhardt, J. Sprenger, J. Petersen, C. Detter, and A. Schlaefer

# Robust Tracking with Particle Filtering for Fluorescent Cardiac Imaging

<https://doi.org/10.1515/cdbme-2025-0315>

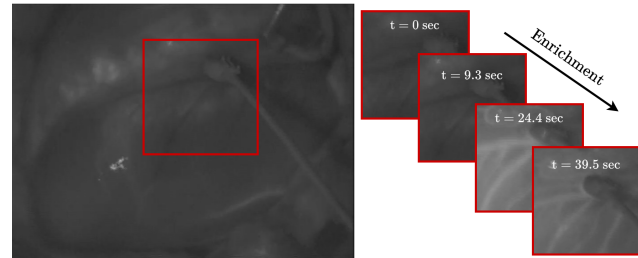
**Abstract:** Intraoperative fluorescent cardiac imaging enables quality control following coronary bypass grafting surgery. We can estimate local quantitative indicators, such as cardiac perfusion, by tracking local feature points. However, heart motion and significant fluctuations in image characteristics caused by vessel structural enrichment limit traditional tracking methods. We propose a particle filtering tracker based on cyclic-consistency checks to robustly track particles sampled to follow target landmarks. Our method tracks 117 targets simultaneously at 25.4 fps, allowing real-time estimates during interventions. It achieves a tracking error of  $(5.00 \pm 0.22 \text{ px})$  and outperforms other deep learning trackers  $(22.3 \pm 1.1 \text{ px})$  and conventional trackers  $(58.1 \pm 27.1 \text{ px})$ .

**Keywords:** Motion Estimation, Heart Surface Tracking, Fluorescent Cardiac Imaging, Deep Learning

## 1 Introduction

Fluorescent cardiac imaging (FCI) offers a non-invasive technique for visualizing coronary vessels. FCI provides real-time, dynamic information about cardiac function to diagnose pathological conditions, such as coronary bypass stenosis [1] or graft assessment [2]. Additionally, clinicians can estimate quantitative values based on fluorescence intensity; for example, to determine myocardial perfusion [1]. To achieve this, robustly tracking local features on the deforming and moving heart surface presents a significant challenge and remains highly desirable.

Conventional markerless object tracking methods, such as Minimum Output Sum of Squared Error (MOSSE) [3] and Kernelized Correlation Filter (KCF) [4], operate with computational efficiency and enable real-time tracking. However, these methods deliver limited tracking accuracy when objects



**Fig. 1: Fluorescent Cardiac Imaging:** Video sequences illustrate different stages of the heart enrichment procedure. Large motions and varying image features challenge the effectiveness of conventional tracking approaches.

experience partial or full occlusion, images contain artifacts (e.g., motion blur caused by rapid movements), or image features change (e.g., during contrast enrichment).

Recently, tracking any point (TAP) approaches [5, 6] have gained attention due to its ability to estimate both dense and long-range pixel-level trajectories in video sequences, even with occlusions. These techniques fundamentally use a coarse-to-fine tracking design that iteratively predicts an initial "coarse" point trajectories using low-resolution features and then refines it using local, spatiotemporal information at higher resolution. Researchers train these methods either with unlabeled raw data combined with unsupervised or semi-supervised learning [6], or with large-scale synthetic data through supervised learning [5].

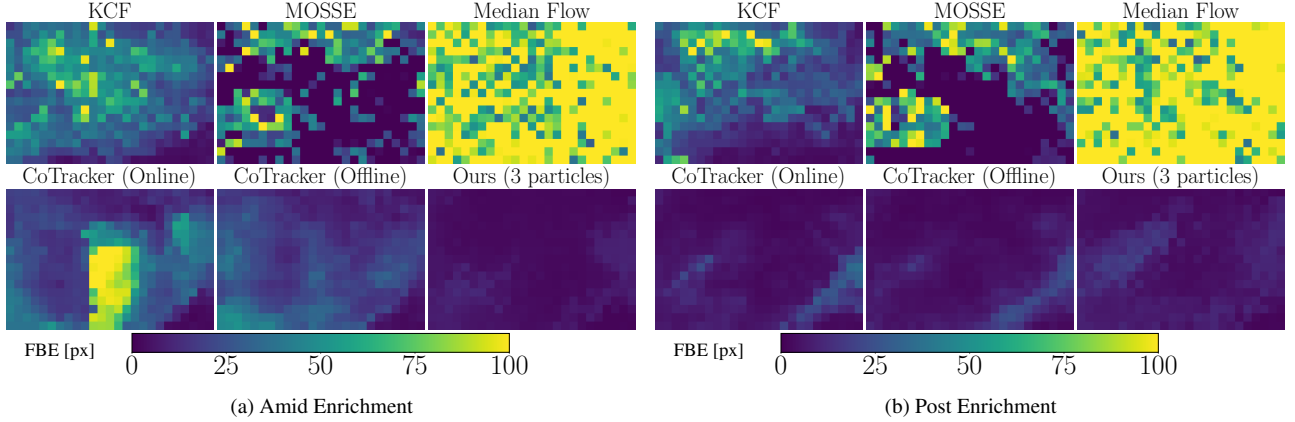
In this work, we propose to retain the benefits of the TAP algorithm, trained on a large-scale dataset, while achieving real-time tracking performance without fine-tuning by refining its predictions using a particle filter-style filtering mechanism. Our study shows that the proposed method outperforms both conventional and deep learning trackers and exhibits greater robustness to large appearance and scale changes during both amid enrichment (AE) and post enrichment (PE) scenarios.

## 2 Method

### 2.1 Notations

Given a video  $\{\mathcal{I}_t\}_{t=1}^T$ , which consists of a sequence of  $T$  frames  $\mathcal{I}_t \in \mathbb{R}^{H \times W \times 3}$ , and an anchor point  $\mathcal{Q} = (t^q, x^q, y^q) \in \mathbb{R}^3$ , where  $t^q$  denoting the query frame index, and  $(x^q, y^q)$  representing the corresponding spatial location of

\*Corresponding author: S. Guttikonda, M. Neidhardt, J. Sprenger, A. Schlaefer, Institute of Medical Technology and Intelligent Systems, Hamburg University of Technology, Am Schwarzenberg-Campus 3, Hamburg, Germany, e-mail: [suresh.guttikonda@tuhh.de](mailto:suresh.guttikonda@tuhh.de)  
J. Petersen, C. Detter, Department of Cardiovascular Surgery, University Heart and Vascular Center Hamburg, Hamburg, Germany  
A. Schlaefer, SustAlnLivWork Center of Excellence, Kaunas, Lithuania



**Fig. 2: Tracking Error Dissemination:** We present the distribution of the FBE for all evaluated trackers. Tracking occurs (a) during contrast agent injection - amid enrichment (AE), and (b) post enrichment (PE), on video sequences with a duration of 15 s, respectively.

the anchor point, the point tracking problem aims to estimate the trajectory of this anchor point across the video. Specifically, the forward point track comprises predictions on subsequent frames of the image sequence and is formally denoted as  $\mathcal{P}_t^f = (x_t, y_t) \in \mathbb{R}^2, t \in [1, T]$ , with  $(x_t, y_t) = (x^q, y^q)$ .

Correspondingly, we define a backward point track consisting of predictions performed from the last to the first frame of the sequence. It is formally denoted as  $\mathcal{P}_t^b = (\hat{x}_t, \hat{y}_t) \in \mathbb{R}^2, t \in [1, T]$ , with  $(\hat{x}_T, \hat{y}_T) = (x_T, y_T)$ .

## 2.2 Our Tracking Algorithm

We propose to track the center of a bounding box with size  $w \times w$  pixels. First, we sample  $M$  points, referred to as particles,  $\{\mathbf{X}_0^m\}_{m=1}^M \in \mathbb{R}^2$ , from a Gaussian distribution  $\mathcal{N}(\mathbf{X}_0 = [x^q, y^q]^\top; \sigma_0 = 5.0)$  centered at the bounding box center. Each particle is initialized with an unbiased uniform weight, such that  $\{w_0^m\}_{m=1}^M = \frac{1}{M}$  for  $m = 1, \dots, M$ , with the weights satisfying  $\sum_{m=1}^M w_t^m = 1$  for  $t \in [0, T]$ .

Next, we compute forward prediction tracks for the sampled particles across a sequence of  $L$  filtering window’s video frames. We recognize that particles experience varying tracking accuracy due to the highly deformable heart surface, which causes occlusions and changes in image features from contrast enrichment. To maintain robustness, we remove particles based on their corresponding weights. Following this, we estimate the Gaussian likelihood ( $\mathcal{L}$ ) for each particle as,

$$\mathcal{L}(\mathcal{P}_t^b, \Sigma | \mathcal{P}_t^f) = \frac{1}{2\pi\sqrt{|\Sigma|}} \exp\left(-\frac{1}{2}(\mathcal{P}_t^b - \mathcal{P}_t^f)^\top \Sigma^{-1}(\mathcal{P}_t^b - \mathcal{P}_t^f)\right) \quad (1)$$

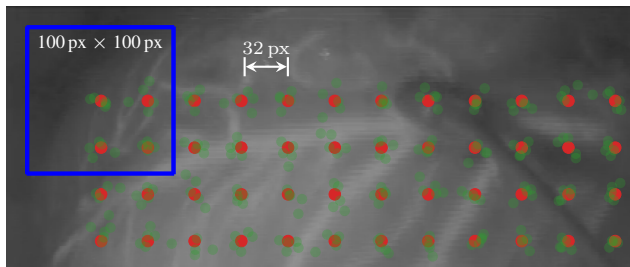
with the covariance matrix  $\Sigma = \begin{bmatrix} \sigma^2 & 0 \\ 0 & \sigma^2 \end{bmatrix}$ , the standard deviation  $\sigma = 3.0$ ,  $d$  represent the data dimensions, and  $|\Sigma|$  denote the determinant of the covariance matrix.

At the beginning of  $l^{\text{th}}$  filtering window step, each particle’s weight, from previous  $l - 1^{\text{th}}$  step, is updated using their corresponding likelihoods (see Equation 1). Finally, we perform stochastic universal soft-resampling<sup>1</sup> [7, 8] with a trade-off resampling parameter  $\alpha = 0.5$ . This process retains or replaces particles with lower weights by generating new particles sampled near those with higher weights at the end of each filtering window, thereby enhancing tracking robustness and accuracy.

## 2.3 Experimental Evaluation

We evaluate different tracking approaches using 14 videos recorded during pig heart fluorescent cardiac interventions. Following anesthesia, the pigs undergo coronary revascularization on their beating hearts and median sternotomy. During the procedure, we inject a contrast agent (indocyanine green) and expose the heart to near-infrared light at 785 nm. The fluorescence signal is captured using an FSI device (LLS GmbH, Ulm, Germany) equipped with band-pass filters that block excitation light while selectively transmitting fluorescent light at 830 nm. Each recording lasts approximately 100 s, with a temporal resolution of 25 Hz and a spatial resolution of 512 px  $\times$  384 px. Figure 1 illustrates a complete enrichment cycle, including both amid and post enrichment sequences. We conduct all training and evaluations using a GPU (Titan RTX, Nvidia, California, USA).

<sup>1</sup> <https://github.com/stanford-iprl-lab/torchfilter>



**Fig. 3: Evaluation:** We evaluate tracking performance using bounding boxes (blue) arranged in a grid structure. The red dots represent anchor points positioned at the center of each bounding box. For our proposed tracker, we sample particles (green dots) located within the bounding boxes.

To evaluate tracking performance, we perform cyclic-consistency checks between the forward and backward predicted point tracks [9]. We define the forward-backward tracking error (FBE) for each anchor point as follows:

$$FBE = \sum_{t=0}^T \|\mathcal{P}_t^b - \mathcal{P}_t^f\|_2 \quad (2)$$

where  $\|\cdot\|_2$  is the Euclidean norm.

As shown in Figure 3, we estimate tracking performance on a grid of anchor points spaced 32 px apart along both axes. We place the sample anchor points at the center of bounding boxes with size  $w=100$  px, as smaller sizes result in immediate tracking failures. For comparison, we select conventional bounding box-based trackers<sup>2</sup> including MOSSE [3], KCF [4], and Median Flow [9]. For our proposed tracker, forward and backward tracking is performed using CoTracker’s online and offline, respectively. We evaluate our tracker using  $M \in [3, 5, 25]$  sampled particles and set the filtering window length to  $L = 16$  frames for all experiments.

### 3 Results

First, as shown in Figures 2a and 2b, we assess the distribution of the FBE, defined in Equation 2, on the surface of the heart during amid-enrichment (AE) and post-enrichment (PE) sequences, respectively. The results reveal that traditional trackers (top row) exhibit a non-homogeneous FBE distribution; for example, MOSSE tracks only discrete features on the heart surface with low FBE. In PE tracking, deep learning-based trackers provide more consistent estimates, whereas they only partially succeed during AE tracking (see CoTracker online). Our approach delivers robust estimates that remain consistent regardless of tracking location, achieving a mean FBE of

$4.9 \pm 4.8$  px and  $4.4 \pm 3.6$  px for AE and PE sequences, respectively.

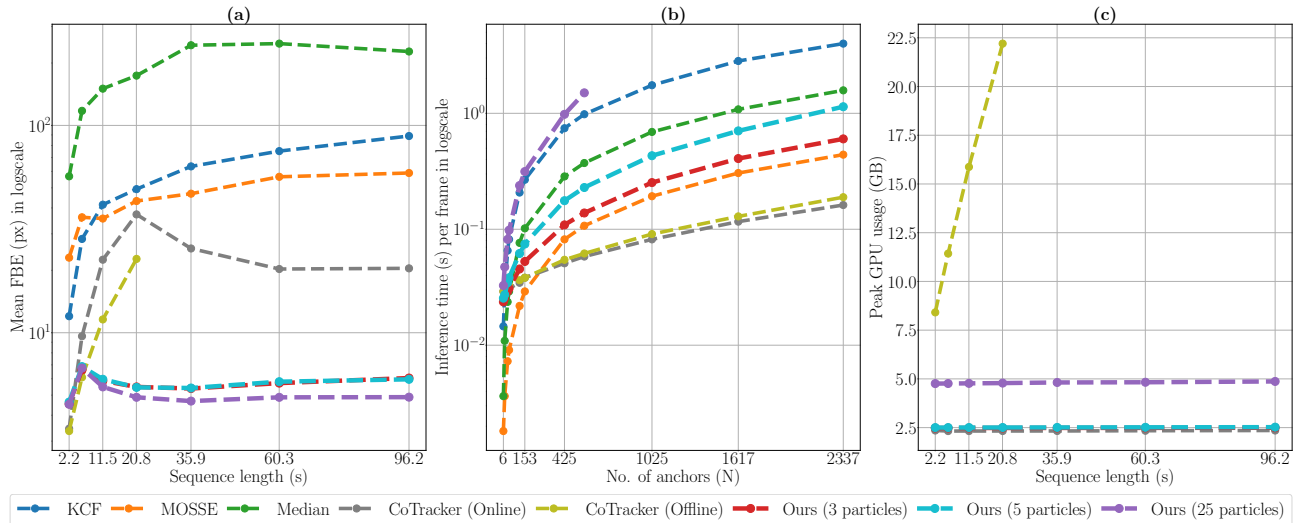
**Tab. 1: Tracking Performance:** We report the mean FBE and frames per second (fps) for tracking 117 anchor points across 14 videos during contrast agent injection, amid enrichment (AE), and post enrichment (PE), each with a tracking duration of 15 s. We refer to the total tracking duration of 90 s, which includes both AE and PE sequences, as "Total".

Tracker	fps	FBE		
		AE	PE	Total
KCF	5.2	$24.3 \pm 13.7$	$26.0 \pm 17.3$	$58.1 \pm 27.1$
MOSSE	<b>50.8</b>	$38.3 \pm 17.9$	$43.8 \pm 21.0$	$70.8 \pm 17.8$
Median Flow	13.9	$84.7 \pm 49.9$	$83.5 \pm 47.9$	$159.9 \pm 59.2$
CoTracker (Online)	42.5	$25.8 \pm 19.9$	$6.4 \pm 3.4$	$22.3 \pm 11.0$
CoTracker (Offline)	31.3	$15.4 \pm 10.6$	$5.7 \pm 3.4$	–
Ours (3 particles)	<b>25.4</b>	$4.9 \pm 4.8$	$4.4 \pm 3.6$	$5.0 \pm 2.2$
Ours (5 particles)	17.9	$4.8 \pm 4.6$	$4.3 \pm 3.4$	$4.8 \pm 2.1$
Ours (25 particles)	4.2	<b><math>4.5 \pm 4.7</math></b>	<b><math>3.9 \pm 3.3</math></b>	<b><math>4.3 \pm 2.2</math></b>

Additionally, we calculate the average FBE for image sequences of varying durations, as shown in Figure 4a. Compared to other trackers, our approach achieves a notably lower mean FBE for tracking sequences longer than 10 s. Figure 4a demonstrates that while our tracking error remains consistently below 7 px during AE, the mean FBE significantly increases for all other trackers. During PE, the error varies only marginally across all methods. A similar trend appears in Table 1, where conventional trackers show overall higher tracking errors compared to our method. Notably, unlike our approach, CoTracker’s tracking error increases only slightly during PE sequences but nearly triples during AE sequences. Furthermore, increasing the number of particles reduces the FBE in our approach. With just three particles, our method outperforms all other trackers, achieving an overall FBE of  $5.0 \pm 2.2$  px.

Compared to CPU-based traditional trackers, as shown in Figure 4b, our approach exhibits a longer inference time. However, the inference time per anchor point decreases as we reduce the number of particles. Achieving a frame rate of up to 25.4 fps with three particles—matching the fluorescent cardiac image stream—we can track 117 anchor points in parallel while maintaining a lower FBE compared to other trackers. Figure 4c presents the GPU peak consumption for various tracking sequence lengths by deep learning-based trackers. While our method remains stable over longer sequences, it incurs slightly higher computational costs than CoTracker (on-

<sup>2</sup> [https://docs.opencv.org/3.4.20/d0/d0a/classcv\\_1\\_1Tracker.html](https://docs.opencv.org/3.4.20/d0/d0a/classcv_1_1Tracker.html)



**Fig. 4: Tracking Performance of All Trackers:** (a) Tracking accuracy vs sequence length, (b) Inference time vs the number of anchor points, and (c) Peak GPU memory usage vs sequence length.

line) due to the use of  $M$  particles per anchor point, although it remains more efficient than CoTracker (offline).

## 4 Discussion and Conclusion

Real-time capture of rapid movements and changes on the heart’s surface is essential for estimating quantitative metrics and assessing heart functions. While conventional trackers offer computational efficiency, they exhibit poor tracking accuracy during fluorescent cardiac imaging (FCI). Advanced deep learning–based trackers, such as CoTracker, demonstrate improved tracking performance with robustness to occlusions; however, they still struggle to overcome domain gaps and face trade-offs between accuracy and GPU computational constraints over longer sequences. In this work, we employ a particle filter mechanism to refine CoTracker’s predictions, enabling robust real-time estimates even when image features change over time, such as during contrast agent injection.

Our experiments indicate that, when used out-of-the-box, CoTracker (offline) outperforms conventional trackers but is limited to video sequences of up to 21 s. In contrast, CoTracker (online) can handle longer sequences but achieves lower overall performance. By applying our proposed approach, we achieve a fourfold improvement in tracking performance on 90.0 s-long video sequences while simultaneously tracking 117 anchor points at a temporal sampling rate of 25.0 fps.

### Author Statement

This research was co-funded by the MARLOC project (DFG, grant SCHL 1844-10-1) and by the European Union under Horizon Europe programme grant agreement No. 101059903;

and by the European Union funds for the period 2021-2027. Conflict of interest: none. Informed consent: obtained. Ethical approval: not applicable.

## References

- [1] Detter, C., Russ, D., Kersten, J.F., Reichenspurner, H., Wipper, S.. Qualitative angiographic and quantitative myocardial perfusion assessment using fluorescent cardiac imaging during graded coronary artery bypass stenosis. *Int J Cardiovasc Imaging* 2018;.
- [2] Ohmes, L.B., Di Franco, A., Di Giammarco, G., Rosati, C.M., Lau, C., Girardi, L.N., et al. Techniques for intraoperative graft assessment in coronary artery bypass surgery. *Journal of Thoracic Disease* 2017;.
- [3] Bolme, D.S., Beveridge, J.R., Draper, B.A., Lui, Y.M.. Visual object tracking using adaptive correlation filters. In: *CVPR*. IEEE; 2010;.
- [4] Henriques, J.F., Caseiro, R., Martins, P., Batista, J.P.. Exploiting the circulant structure of tracking-by-detection with kernels. In: *ECCV*. Springer; 2012;.
- [5] Doersch, C., Yang, Y., Vecerik, M., Gokay, D., Gupta, A., Aytar, Y., et al. TAPIR: tracking any point with per-frame initialization and temporal refinement. In: *ICCV*. IEEE; 2023;.
- [6] Karaev, N., Makarov, I., Wang, J., Neverova, N., Vedaldi, A., Ruppert, C.. Cotracker3: Simpler and better point tracking by pseudo-labelling real videos. *CoRR* 2024;.
- [7] Thrun, S., Burgard, W., Fox, D.. Probabilistic robotics. *Intelligent robotics and autonomous agents*; MIT; 2005.
- [8] Karkus, P., Hsu, D., Lee, W.S.. Particle filter networks with application to visual localization. In: *CoRL*. PMLR; 2018;.
- [9] Kalal, Z., Mikolajczyk, K., Matas, J.. Forward-backward error: Automatic detection of tracking failures. In: *ICPR*. IEEE; 2010;.