Tracking Everything in Robotic-Assisted Surgery

Bohan Zhan[1], Wang Zhao[2], Yi Fang[3], Bo Du[4], Francisco Vasconcelos[5], Danail Stoyanov[5], Daniel S. Elson[1], Baoru Huang[1,5,6]

***Abstract*— Accurate tracking of tissues and instruments in videos is crucial for Robotic-Assisted Minimally Invasive Surgery (RAMIS), as it enables the robot to comprehend the surgical scene with precise locations and interactions of tissues and tools. Traditional keypoint-based sparse tracking is limited by featured points, while flow-based dense two-view matching suffers from long-term drifts. Recently, the Tracking Any Point (TAP) algorithm was proposed to overcome these limitations and achieve dense accurate long-term tracking. However, its efficacy in surgical scenarios remains untested, largely due to the lack of a comprehensive surgical tracking dataset for evaluation. To address this gap, we introduce a new annotated surgical tracking dataset for benchmarking tracking methods for surgical scenarios, comprising real-world surgical videos with complex tissue and instrument motions. We extensively evaluate state-of-the-art (SOTA) TAP-based algorithms on this dataset and reveal their limitations in challenging surgical scenarios, including fast instrument motion, severe occlusions, and motion blur, etc. Furthermore, we propose a new tracking method, namely SurgMotion, to solve the challenges and further improve the tracking performance. Our proposed method outperforms most TAP-based algorithms in surgical instruments tracking, and especially demonstrates significant improvements over baselines in challenging medical videos. Our code and dataset are available at https://github.com/zhanbh1019/SurgicalMotion.**

I. INTRODUCTION

Robotic-Assisted Minimally Invasive Surgery (RAMIS) is widely applied in various types of medical procedure [1]. Compared to traditional open surgery, RAMIS offers advantages such as reducing human errors, enhancing the capability for remote surgery, mitigating pain, and lowering the risk of infection [2]–[4]. The ultimate objective of RAMIS is fully automated surgery, which places high demands on the robot's ability to understand the surgical scene [5], [6]. Therefore, motion tracking plays a vital role in RAMIS. Precise tracking of tissues and surgical instruments enables the robot to locate instruments and target tissues, and assess their relative positions and interactions.

Traditional visual tracking algorithms in surgical scenarios have several limitations. Instrument tracking typically relies on box-based or segmentation-based methods, which struggle to capture detailed rotation and deformation of instruments effectively [7]–[9]. Tissue tracking, on the other hand, often depends on sparse feature matching or dense optical flow methods [10]. However, sparse feature matching performs poorly when handling deformable or weakly-textured tissues, and it only provides tracking on sparse feature points which may not fully fulfill the practical demands [11]. While optical flow is a dense tracking method, it only tracks motion between adjacent frames, making it vulnerable to occlusion and prone to tracking drifts in medical videos [12], [13].

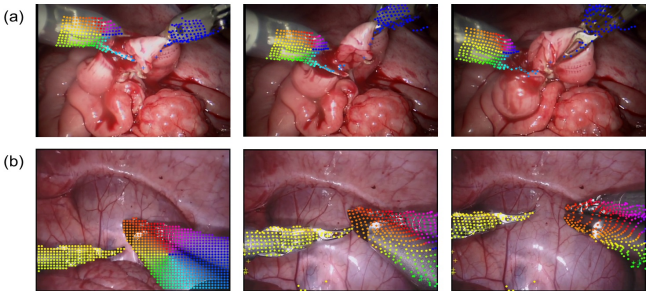

Fig. 1. The demonstration of our method for tracking every point across entire surgical video. (a), (b) present results from different videos, with the tracking of instruments displayed from left to right over time.

In the general field of motion tracking, similar challenges also exist. To address these limitations, recent research has proposed the TAP algorithm [14], which directly tracks every pixel across long-term videos and estimates occlusions via spatial-temporal feature learning and alignment, demonstrating superior performance over traditional tracking methods on a wide array of applications. However, the TAP-based algorithm has not been employed in surgical environments. Most TAP-based methods are trained on synthetic datasets of natural scenes, which include precise point trajectories as ground truth [15]–[19]. The significant differences between natural scenes and surgical environments, such as poor lighting conditions, lack of significant texture features, and high specular reflection, present great challenges for applying TAP-based algorithms in surgical videos.

To evaluate the performance of TAP-based algorithms, several benchmarks have been established, including TAP-Vid [14] and PointOdyssey [17], which feature manually or automatically annotated point trajectories. However, these benchmarks are limited to non-surgical domains, and there is a notable lack of equivalent datasets for surgical videos. In the context of surgical tracking, the SuPer dataset [20] and the SurgT dataset [21] are the most relevant existing datasets. Nevertheless, these datasets have significant limitations: the SuPer dataset has an insufficient number of labeled frames, while the SurgT dataset provides only sparse annotations, with a single labeled point per frame. As a result, existing

[1]The Hamlyn Centre for Robotic Surgery, Imperial College London, SW7 2AZ, UK Baoru.Huang18@imperial.ac.uk
[2]Department of Computer Science, Tsinghua University, Beijing, China
[3]Embodied AI and Robotics Lab, NYU Abu Dhabi, Abu Dhabi, UAE
[4]School of Computer Science, Wuhan University, Wuhan, China
[5]Hawkes Institute, University College London, WC1E 6BT, UK
[6]Department of Computer Science, University of Liverpool, L69 7ZX, UK

datasets fall short in providing a comprehensive evaluation of TAP-based algorithms.

To address this gap, we developed a new dataset with manually labeled point trajectories to evaluate tracking algorithms in surgical scenarios. Specifically, we collected 20 real-world surgical videos, each with approximately 60 frames, and manually labeled 25 points per frame. Leveraging this dataset, we conducted a comprehensive evaluation of existing TAP-based algorithms, exposing their limitations in tracking fast-moving instruments in robotic surgery and other challenging regions. To further address these challenges, we designed an effective tracking method, SurgMotion, building upon OmniMotion [22] and proposed key constraints, including tool mask constraint, as-rigid-as-possible (ARAP) constraint, and sparse feature matching guidance. These innovations collectively yield significant improvements in tracking accuracy, particularly in challenging surgical videos. This method tracks any points, enabling simultaneous tracking of multiple surgical instruments without the need to distinguish between instrument categories, making it highly applicable to various surgical scenarios. To summarize, the key contributions of this paper are as follows:

- We established a new surgical video tracking dataset, pioneering the exploration of tracking algorithms in real-world surgical scenarios.
- We benchmarked existing TAP-based algorithms on our dataset, revealing their limitations in tracking complex surgical motions in robotic surgery.
- We proposed a new tracking method, SurgMotion, for accurate tracking in surgical videos. Extensive experiments demonstrate that our method outperforms existing TAP-based methods and shows substantial improvements in challenging surgical videos.

II. Related Work

A. Tracking Datasets in Surgical Domain

Generating accurate tracking ground truth is highly challenging, and as a result, there is currently a lack of datasets for evaluating tracking algorithms in surgical scenarios. Existing datasets for tissue tracking, such as Semantic SuPer [23] and STIR [24], use bead markers and indocyanine green to label real tissues. However, their precision is insufficient for accurately assessing single-point trajectories. The SuPer [20] and SurgT [21] datasets, which are the closest to the requirements of TAP-based algorithms, offer manually annotated points as ground truth. However, the SuPer dataset contains only 52 frames with 20 points per frame, while SurgT, despite having 24,548 frames, provides just one annotated point per frame. Therefore, both the scale and the annotation accuracy of these datasets are inadequate for evaluating the performance of TAP-based algorithms.

B. Surgical Scene Tracking

Vision-based tracking in surgical scenarios is generally divided into tissue tracking and surgical instrument tracking. Tissue tracking relies on sparse feature matching or dense optical flow analysis. For example, a CNN-based optical flow method [12] fine-tunes FlowNet [25], enabling a fast convolutional model for tissue tracking. The KINFlow [26] uses k-nearest keypoint correspondences to track tissue motion. Additionally, SENDD [27] employs Graph Neural Networks to match sparse keypoints and estimate per-point depth and 3D flow. Surgical instrument tracking is typically solved by box-based or segmentation-based algorithms. For example, YOLO-based tracking methods [28], known for their real-time performance, locate instruments using bounding boxes. However, segmentation-based approaches offer higher accuracy. Transformer models like TraSeTR [29], and MATIS [30] demonstrate high precision in instrument tracking. Recently, foundation models, such as MedSAM [31] and SurgicalSAM [32], further enhance the accuracy of surgical instrument segmentation and tracking.

C. Tracking Any Point

Tracking any point algorithms have been explored in recent years. PIPs [16] pioneers the concept of pixel tracking as a long-range motion estimation problem, enabling the tracking of every point in a video sequence within a small, fixed time window (8 frames). PIPs++ [17] expands the temporal field of view of the original PIPs. Subsequently, TAP-Vid [14] formalizes this problem and introduces a benchmark that includes manually annotated real-world datasets as well as a large set of synthetic datasets. TAP-Vid also proposes a simple baseline method, TAP-Net, which is combined with PIPs to create TAPIR [15]. MFT [18] tracks every pixel in a template based on the combination of the optical flow field and different time spans. CoTracker [19] presents a powerful transformer-based model to track points in the video by accounting for the correlation of points and tracking them jointly. Unlike the aforementioned methods trained on ground truth point trajectories from synthetic datasets, OmniMotion [22] introduces a test-time optimization algorithm that requires no prior knowledge, achieving accurate, full-length motion estimation of every pixel in a video.

III. Methodology

In this work, we developed a surgical dataset with manually labeled point trajectories as the ground truth to evaluate surgical tracking algorithms. Additionally, we proposed a new method, SurgMotion, to adapt the existing TAP-based algorithm to improve its performance in surgical instrument tracking. Our approach incorporates three key loss functions: (1) a mask loss that constrains points within the instrument area, (2) an ARAP loss that enforces accurate point correspondence, and (3) a long-term loss that improves tracking accuracy across distant frames. The overview of our method can be seen in Fig. 3.

A. Dataset Creation

1) Data Sources: Our dataset is derived from the SurgT benchmark [21] and the EndoNerf dataset [33]. The videos in the SurgT benchmark originate from the Hamlyn dataset [34], the SCARED dataset [35], and the Kidney boundary dataset [36]. These videos cover a wide range of surgical

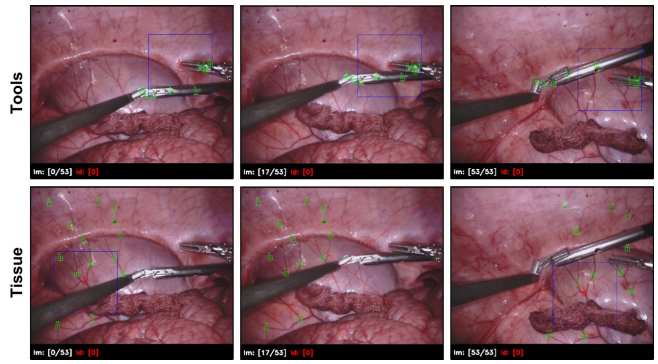

Fig. 2. Examples of the dataset, where tissues and surgical tools are annotated separately. The red cross indicates that this point is occluded in this frame.

scenes, providing a comprehensive evaluation of the algorithms, including da Vinci robotic prostatectomy, *in-vivo* porcine abdominal kidney procedures, and *in-vivo* human surgeries, specifically robotic-assisted partial nephrectomy.

2) Data Processing: From the aforementioned datasets, we selected video clips that contain both tissues and instruments, trimming them to approximately 50 to 70 frames each. Videos with an original resolution of 1280×1024 pixels were downsampled to 640×512 pixels, while the remaining videos, originally at 640×480 pixels, retained their original resolution

3) Annotation Process: To evaluate the tracking algorithms, we manually annotated the selected videos. The annotation tool was developed by adapting the SurgT-labelling tool to meet our specific requirements. Our manual annotation process basically followed the widely-used TAP-Vid-DAVIS dataset guidelines [14].

For each video, we annotated 25 points per frame, consistent with the TAP-Vid-DAVIS dataset [14]. Surgical instruments and tissues were labeled separately. Typically, each video contained two moving instruments, with 5 points annotated per instrument, totaling 10 points. The remaining 15 points were allocated for tissue annotations.

For surgical instruments, we selected points at the instrument tips, corners, or locations with distinctive features (e.g., screws, mechanical joints). For tissues, points were chosen in areas with textures, such as blood vessel junctions. Each selected landmark was manually tracked across all frames, with occluded points labeled as "occluded" and points that moved out of the frame as "out of view". Fig. 2 provides a visual example of the annotated dataset.

B. Preliminaries

OmniMotion [22] introduces a test-time optimization method for long-term pixel-wise tracking, even under occlusion. It represents videos as a canonical 3D volume G and tracks pixels via bijections between local frames and the canonical frame. Specifically, OmniMotion [22] consists of three main steps: first, the 2D point p_i is sampled along the ray orthogonal to the image plane, lifting the 2D point into a 3D point x_i. Then, bijections are established between the 3D point x_i in the local frames and a point u in the canonical volume, represented as $u = T_i(x_i)$. These bijections are parameterized as invertible neural networks Real-NVPs [37]. Using these bijections, the 3D point can be mapped from one local frame L_i to another local frame L_j:

$$x_j = T_j^{-1} \circ T_i(x_i) \tag{1}$$

Following the NeRF [38] approach, OmniMotion [22] maps all 3D points $u \in G$ in the canonical volume to color and density using a coordinate-based MLP network F, which is represented as:

$$(\sigma_k, c_k) = F_\theta(M_\theta(x_i^k; \psi_i)) \tag{2}$$

Finally, OmniMotion [22] projects the 3D points onto a 2D plane through volume rendering. Specifically, after applying the bijections, a set of 3D points x_i^k corresponding to p_i in frame i are mapped to the canonical volume u, and then mapped to frame j to obtain another set of 3D points x_j^k. These 3D points are aggregated via alpha compositing to produce the 3D point $\hat{x}_j$

$$\hat{x}_j = \sum_{k=1}^{K} T_k \alpha_k x_j^k, \quad \text{where} \quad T_k = \prod_{l=1}^{k-1}(1-\alpha_l) \tag{3}$$

By projecting the 3D point x_j onto the 2D plane, the 2D point p_j is obtained. The same process can also be applied to obtain the image space color C_j. The model is supervised by an optical flow loss and an RGB loss, where the predicted flow is guided by the optical flow calculated using the pre-trained RAFT model [39].

C. Proposed Method: SurgMotion

In surgical scenarios, the motion patterns of the instruments and tissues exhibit two distinctly different characteristics. Tissues tend to undergo significant deformation but generally move slowly when manipulated with surgical instruments, making point tracking relatively straightforward. In contrast, instruments move rapidly, often resulting in motion blur in the video. Their thin and fine tips further complicate point tracking, frequently leading to substantial loss of points during the tracking process. To mitigate these challenges, we incorporate three specialized loss functions for surgical instruments to enhance tracking performance.

Tool Mask Constraints. An intuitive approach to improve instrument tracking is to ensure that points on instruments stay within their designated regions throughout the tracking process. To achieve this, we utilize existing segmentation models [31] to extract masks of the surgical instruments and introduce a mask loss to ensure that the points on the instruments consistently remain within the instrument's mask during training. This strategy helps to prevent issues where tracking points on surgical instruments are erroneously "left behind" on the tissue. The mask loss was defined as follows:

$$\mathcal{L}_{mask} = \sum_{x_i \in \Omega_M^i} \|M(x_i) - M(x_j)\|_2^2 \tag{4}$$

where Ω_M^i represents the set of all pixels within the mask in frame i. M is a binary image where 1 indicates the regions

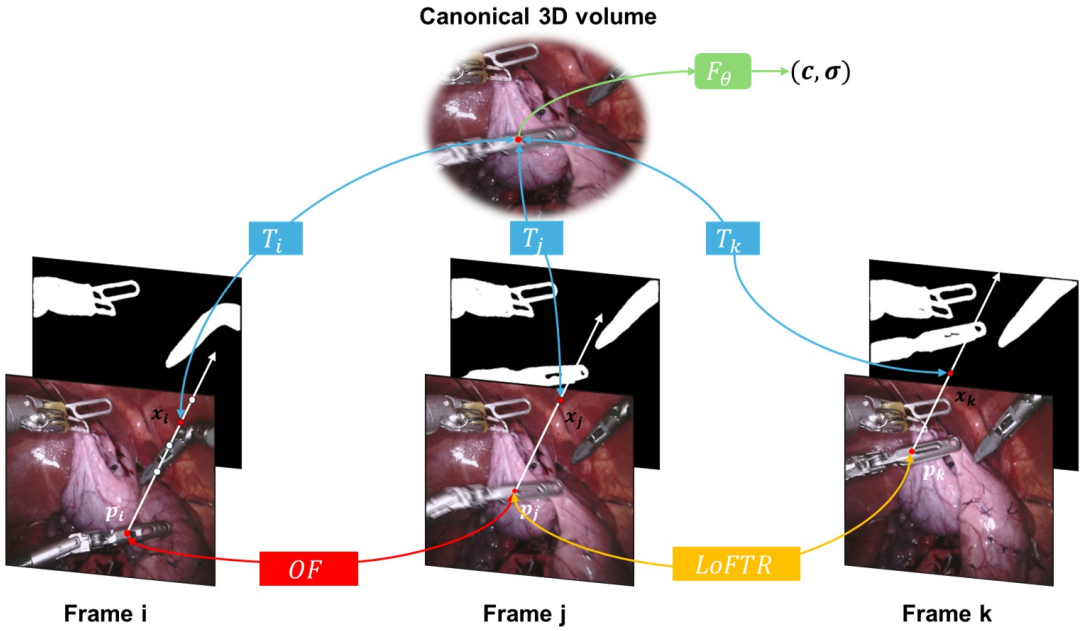

Fig. 3. Method overview: First, 2D points p_i are lifted to 3D points x_i. Then, by using a bijective transformation T_i, the x_i in the local frame are mapped to a canonical 3D volume as u, and subsequently mapped to another local frame through an inverse bijection. A coordinate-based network F_θ is employed to compute the corresponding color c and density σ of point u in the canonical volume, with the 2D positions obtained through alpha compositing. To ensure that points on the tools remain correctly mapped to the tool area, we introduce a tool mask and ARAP constraints. Additionally, since OmniMotion [22] is supervised by optical flow (OF), which becomes inaccurate in distant frames, we incorporate LoFTR [40] feature matching to enhance long-term tracking capabilities.

occupied by the tools. x_j is the corresponding point of x_i in frame j. The mask loss is minimized by reducing the mean squared error (MSE) between masks across frames, thereby ensuring that points belonging to instruments remain consistently within the instrument mask over time.

As Rigid As Possible Constraints. After applying the Tool Mask Constraints, the model effectively keeps tool points within the mask region. However, this does not guarantee that the points are correctly positioned relative to each other. Given that OmniMotion [22] lifts the points from the local frame to 3D during training, the 3D coordinates of these points can be accessed. With this 3D information, we can establish ARAP constraints because surgical instruments are typically rigid bodies, and the displacement of points on the same instrument are supposed to be consistent. By leveraging this property, we can enhance spatial consistency, ensuring that the points on the tools appear in their correct corresponding positions.

To enforce the ARAP constraint, the first step is to check whether the points belong to the same rigid part of the surgical tool. We employ the K-means clustering algorithm here to classify the points into different motion groups and the ARAP loss is formulated as follows:

$$\mathcal{L}_{arap} = \sum_{(x^k, x^p) \in \Omega_{k,p}} \left\| d(x_i^k, x_j^k) - d(x_i^p, x_j^p) \right\|_1 \quad (5)$$

where $\Omega_{k,p}$ represents the set of all pairwise points within the same rigid part, and $d(\cdot, \cdot)$ denotes the Euclidean distance function. By minimizing the ARAP loss, the displacement between any pair of corresponding points within the same rigid cluster is enforced to be consistent.

Sparse Feature Matching Guidance. Even with the application of mask loss and ARAP loss, our method still encounters significant challenges in maintaining accuracy during long-term tracking. This is primarily because the method relies on optical flow for supervision and optical flow tends to result in numerous erroneous matches between distant frames. While OmniMotion [22] attempts to filter out these incorrect matches using cycle consistency and appearance filtering, this approach usually leads to the loss of tracking information for many points over long-term tracking, particularly those on surgical instruments. To overcome this limitation, we incorporate a feature matching algorithm to guide long-term tracking. Specifically, we use the LoFTR [40], a transformer-based semi-dense feature matching method, to extract pixel-wise matches from any two frames. By integrating LoFTR [40] into our tracking framework, the information for points that are previously filtered out is successfully recovered. This approach not only enhances the quality of point information available for long-term tracking but also provides additional corresponding points for surgical instruments. We formulate the long-term loss as follows:

$$\mathcal{L}_{long} = \sum_{(p_{i,j} \in \Omega)} \left\| (\hat{p_j} - p_i) - (p_j - p_i) \right\|_1 \quad (6)$$

where Ω is the set of all pairwise points in feature matching. $\hat{p_j}$ is the point matched to p_i by LoFTR [40], and p_j is the corresponding point predicted by our model. We minimize the mean absolute error (MAE) between the predicted matching from our model and the ground truth matching generated by LoFTR [40].

IV. Experiments and Results

A. Evaluation Metrics

Following the TAP-Vid benchmark [14], we evaluate the accuracy of tracking and occlusion on our dataset and the evaluation metrics include:

- $< \delta^x_{\text{avg}}$ represents the average position accuracy of visible points, measuring the percentage of predicted points falling within five distance thresholds: 1, 2, 4, 8, and 16 pixels from their ground truth.
- Average Jaccard (AJ) assesses the joint accuracy of the predicted positions and visibility.

$$\text{AJ} = \frac{\text{True positives}}{\text{True positives} + \text{False positives} + \text{False negatives}}$$

  where true positives are points within the distance thresholds of visible ground truth. False positives are points predicted as visible but are occluded or beyond thresholds, and false negatives are visible ground truth predicted as occluded or outside the thresholds.
- Occlusion Accuracy (OA) measures the accuracy of visibility predictions for all points, including both visible and occluded points.

Following the evaluation protocols of OmniMotion, we resize the images to 256×256 during evaluation.

B. Implementation Details

The experiments were conducted on NVIDIA GPU3090 with PyTorch framework and the model was trained with the Adam optimizer for 100k iterations per video sequence. Each training batch includes 256 extracted point pairs from 8 pairs of frames, with 192 points sampled from optical flow and 64 points from feature matching. We assigned specific weights to the loss functions. For iterations 0 to 20k, w_{mask} and w_{arap} were set to 0, and fixed at 1 thereafter. Furthermore, w_{long}, the weight for the long-term loss, was set to 0.3.

C. Comparisons

We compare our method with five SOTA TAP-based algorithms on our dataset, including RAFT [39], PIPs [16], PIPs++ [17], MFT [18], TAPIR [15], CoTracker [19], and OmniMotion [22]. These algorithms serve as baselines for evaluating the performance of our proposed method. However, tracking methods designed specifically for surgical instruments cannot track any points, making them not directly comparable.

1) Quantitative Comparisons: We compare our method with other baselines on our dataset in Table I. In surgical instrument tracking, our method significantly outperforms RAFT [39], PIPs [16], PIPs++ [17], MFT [18], and TAPIR [15]. Furthermore, it achieves SOTA in both AJ and OA metrics. Compared to our most direct competitor, OmniMotion [22], our method demonstrates improvements across all three metrics. Notably, it surpasses the SOTA method, CoTracker [19], by 7.2% in OA metric.

When examining more challenging videos—defined as "challenging cases" where the position accuracy of visible points ($< \delta^x_{avg}$) in the baseline models is below 75%, typically involving fast-moving instruments and motion blur that increase tracking difficulty. We compare the performance of our method with the baseline methods in these scenarios, and the results are shown in Table II. It can be seen that our method significantly outperforms all other baselines in these challenging cases, where rapidly moving instruments appear and blurred motion happens, indicating the superior capability of our method in handling such difficult scenarios and improving the overall instrument tracking performance.

Although our method primarily targets improving tool tracking performance, without specific optimization for tissue tracking, it still maintains a high tissue tracking accuracy comparable to other baseline methods and achieves runner-up results in both AJ and OA metrics with a small margin from the SOTA. This is because tool movement is generally more salient compared to tissue movement. Consequently, our approach achieves significant advancements in challenging tool tracking scenarios while performing on a par with existing methods for tissue tracking.

2) Qualitative Comparisons: We compare our method qualitatively to the baselines in Fig. 4. Our algorithm demonstrates superior stability in instrument tracking compared to MFT and OmniMotion, especially with fast-moving surgical instruments. While baseline methods often result in a large

TABLE I

Quantitative comparison between our SurgMotion method and baselines. The optimal and suboptimal results are shown in **bold** and underlined respectively.

Method	Tools			Tissue		
	AJ ↑	$< \delta^x_{avg}$ ↑	OA ↑	AJ ↑	$< \delta^x_{avg}$ ↑	OA ↑
RAFT [39]	49.7	64.8	91.1	68.5	84.6	87.6
PIPs [16]	54.1	69.2	88.2	58.5	73.8	88.6
PIPs++ [17]	-	68.3	-	-	83.8	-
MFT [18]	56.1	67.3	87.5	78.0	87.7	94.2
TAPIR [15]	60.8	71.2	88.1	73.6	82.3	94.1
CoTracker [19]	62.8	**77.1**	85.1	78.3	86.6	94.1
OmniMotion [22]	62.0	73.3	91.5	**80.3**	**87.9**	**96.9**
SurgMotion (Ours)	**63.0**	74.2	**92.3**	79.8	87.5	96.1

TABLE II

Comparison between Baselines and our SurgMotion method on Challenging Tool Tracking Cases.

Method	Tools		
	AJ ↑	$< \delta^x_{avg}$ ↑	OA ↑
RAFT [39]	44.3	60.7	89.6
SurgMotion	**59.2**	**71.3**	**90.8**
PIPs [16]	48.5	64.3	87.0
SurgMotion	**57.3**	**69.4**	**90.6**
PIPs++ [17]	-	64.3	-
SurgMotion	-	**71.9**	-
MFT [18]	46.0	59.2	83.2
SurgMotion	**56.9**	**69.6**	**89.9**
TAPIR [15]	55.9	66.6	85.1
SurgMotion	**59.1**	**70.7**	**91.6**
CoTracker [19]	49.1	64.8	79.1
SurgMotion	**52.2**	**65.0**	**90.4**
OmniMotion [22]	47.3	61.1	85.7
SurgMotion	**51.7**	**64.7**	**89.0**

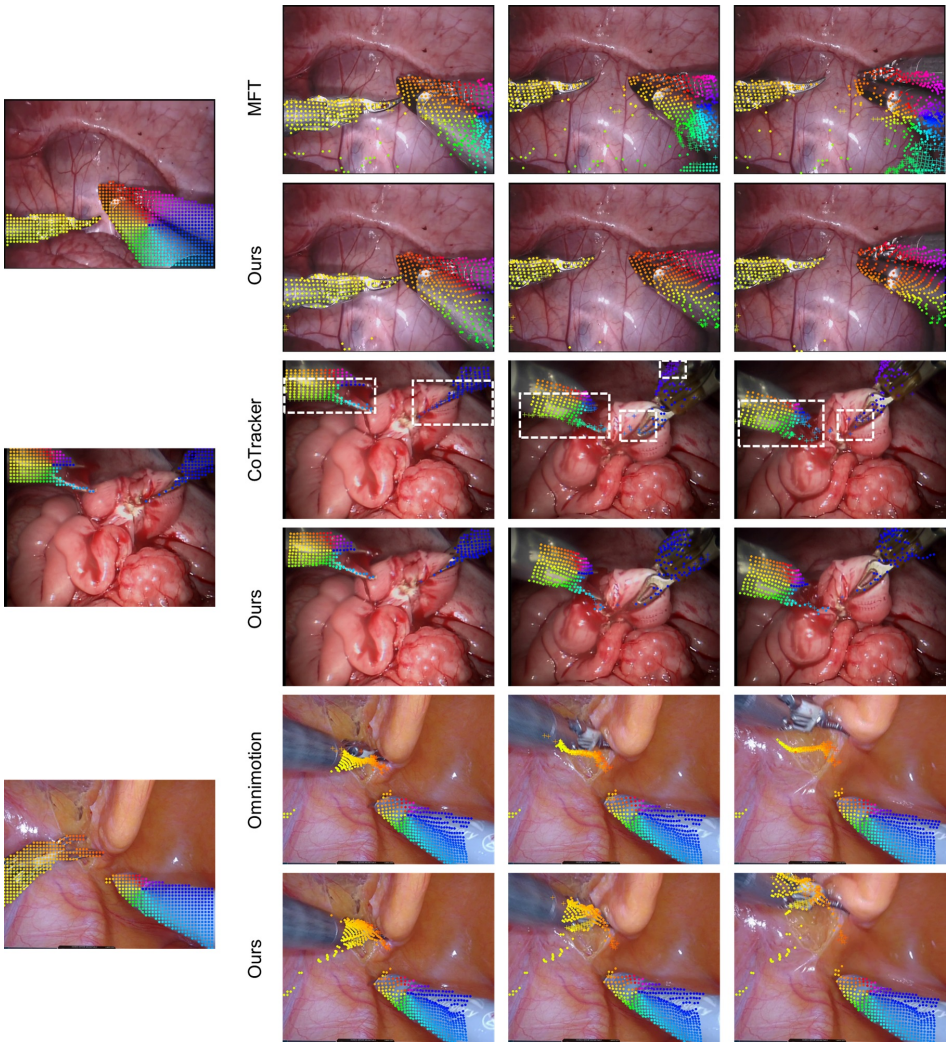

Fig. 4. Qualitative comparison of our method with other baselines on our dataset. The leftmost column shows the initial query points. The three columns on the right display the tracking results over time. Occluded points are marked with a cross "+" and their estimated positions are shown. Notably, the white dashed boxes highlight CoTracker's incorrect occlusion predictions, whereas our method produces accurate results in these cases.

number of lost points on the instruments, our algorithm is more effective at maintaining tracking of these points, thereby avoiding complete failure. Furthermore, compared to CoTracker, our method demonstrates improved performance in occlusion prediction.

D. Ablation Study

We conduct ablation experiments to validate the effectiveness of our method, as shown in Table III. The three proposed loss functions, $\mathcal{L}_{mask}$, $\mathcal{L}_{arap}$ are analyzed, with $\mathcal{L}_{arap}$ applied alongside the mask. We evaluated the individual effects of $\mathcal{L}_{mask}$ and $\mathcal{L}_{long}$, as well as the combined effects of $\mathcal{L}_{mask} + \mathcal{L}_{arap}$ and $\mathcal{L}_{mask} + \mathcal{L}_{long}$. The experimental results demonstrate that both $\mathcal{L}_{mask}$ and $\mathcal{L}_{long}$ individually improve performance. Furthermore, the combination of $\mathcal{L}_{mask}+\mathcal{L}_{arap}$ outperforms $\mathcal{L}_{mask}$ alone while adding $\mathcal{L}_{long}$ produces the superior overall performance compared to the baseline.

TABLE III

ABLATION STUDY OF OUR METHOD.

$\mathcal{L}_{mask}$	$\mathcal{L}_{arap}$	$\mathcal{L}_{long}$	Tools		
			AJ ↑	$< \delta^x_{avg}$ ↑	**OA** ↑
–	–	–	62.0	73.3	91.5
✓	–	–	62.1	73.3	91.5
–	–	✓	62.6	74.0	92.5
✓	✓	–	62.8	74.2	**92.7**
✓	–	✓	62.4	73.8	91.9
✓	✓	✓	**63.0**	**74.2**	92.3

V. CONCLUSIONS

In this paper, we present a new annotated dataset pioneering the evaluation of tracking algorithms in surgical videos, and we comprehensively benchmark SOTA TAP-based algorithms on this dataset. Existing algorithms struggle to track surgical instruments accurately due to their rapid motion, which frequently leads to motion blur and occlusion in the video. To overcome these challenges, we introduce a novel method, SurgMotion, that enhances the tracking performance of surgical instruments, particularly in challenging scenarios, and achieves superior results compared to all tested algorithms.

REFERENCES

[1] P. Fiorini, K. Y. Goldberg, Y. Liu, and R. H. Taylor, "Concepts and trends in autonomy for robot-assisted surgery," *Proceedings of the IEEE*, vol. 110, no. 7, pp. 993–1011, 2022.

[2] M. Diana and J. Marescaux, "Robotic surgery," *Journal of British Surgery*, vol. 102, no. 2, pp. e15–e28, 2015.

[3] J. D. Wright, "Robotic-assisted surgery: balancing evidence and implementation," *Jama*, vol. 318, no. 16, pp. 1545–1547, 2017.

[4] D. Zhang, B. Xiao, B. Huang, L. Zhang, J. Liu, and G.-Z. Yang, "A self-adaptive motion scaling framework for surgical robot remote control," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 359–366, 2018.

[5] M. Yip and N. Das, "Robot autonomy for surgery," in *The Encyclopedia of MEDICAL ROBOTICS: Volume 1 Minimally Invasive Surgical Robotics*. World Scientific, 2019, pp. 281–313.

[6] B. Huang, J.-Q. Zheng, A. Nguyen, C. Xu, I. Gkouzionis, K. Vyas, D. Tuch, S. Giannarou, and D. S. Elson, "Self-supervised depth estimation in laparoscopic image using 3d geometric consistency," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2022, pp. 13–22.

[7] T. Rueckert, D. Rueckert, and C. Palm, "Methods and datasets for segmentation of minimally invasive surgical instruments in endoscopic images and videos: A review of the state of the art," *Computers in Biology and Medicine*, p. 107929, 2024.

[8] B. Huang, A. Nguyen, S. Wang, Z. Wang, E. Mayer, D. Tuch, K. Vyas, S. Giannarou, and D. S. Elson, "Simultaneous depth estimation and surgical tool segmentation in laparoscopic images," *IEEE transactions on medical robotics and bionics*, vol. 4, no. 2, pp. 335–338, 2022.

[9] J. Cartucho, C. Wang, B. Huang, D. S. Elson, A. Darzi, and S. Giannarou, "An enhanced marker pattern that achieves improved accuracy in surgical tool tracking," *Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization*, vol. 10, no. 4, pp. 400–408, 2022.

[10] A. Schmidt, O. Mohareri, S. DiMaio, M. C. Yip, and S. E. Salcudean, "Tracking and mapping in medical computer vision: A review," *Medical Image Analysis*, p. 103131, 2024.

[11] M. C. Yip, D. G. Lowe, S. E. Salcudean, R. N. Rohling, and C. Y. Nguan, "Tissue tracking and registration for image-guided surgery," *IEEE transactions on medical imaging*, vol. 31, no. 11, pp. 2169–2182, 2012.

[12] S. Ihler, M.-H. Laves, and T. Ortmaier, "Patient-specific domain adaptation for fast optical flow based on teacher-student knowledge transfer," *arXiv preprint arXiv:2007.04928*, 2020.

[13] B. Huang, Y.-Y. Tsai, J. Cartucho, K. Vyas, D. Tuch, S. Giannarou, and D. S. Elson, "Tracking and visualization of the sensing area for a tethered laparoscopic gamma probe," *International Journal of Computer Assisted Radiology and Surgery*, vol. 15, no. 8, pp. 1389–1397, 2020.

[14] C. Doersch, A. Gupta, L. Markeeva, A. Recasens, L. Smaira, Y. Aytar, J. Carreira, A. Zisserman, and Y. Yang, "Tap-vid: A benchmark for tracking any point in a video," *Advances in Neural Information Processing Systems*, vol. 35, pp. 13 610–13 626, 2022.

[15] C. Doersch, Y. Yang, M. Vecerik, D. Gokay, A. Gupta, Y. Aytar, J. Carreira, and A. Zisserman, "Tapir: Tracking any point with per-frame initialization and temporal refinement," *arXiv preprint arXiv:2306.08637*, 2023.

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

[17] Y. Zheng, A. W. Harley, B. Shen, G. Wetzstein, and L. J. Guibas, "Pointodyssey: A large-scale synthetic dataset for long-term point tracking," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 19 855–19 865.

[18] M. Neoral, J. Šerỳch, and J. Matas, "Mft: Long-term tracking of every pixel," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2024, pp. 6837–6847.

[19] N. Karaev, I. Rocco, B. Graham, N. Neverova, A. Vedaldi, and C. Rupprecht, "Cotracker: It is better to track together," *arXiv preprint arXiv:2307.07635*, 2023.

[20] Y. Li, F. Richter, J. Lu, E. K. Funk, R. K. Orosco, J. Zhu, and M. C. Yip, "Super: A surgical perception framework for endoscopic tissue manipulation with surgical robotics," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2294–2301, 2020.

[21] J. Cartucho, A. Weld, S. Tukra, H. Xu, H. Matsuzaki, T. Ishikawa, M. Kwon, Y. E. Jang, K.-J. Kim, G. Lee, *et al.*, "Surgt challenge: Benchmark of soft-tissue trackers for robotic surgery," *Medical image analysis*, vol. 91, p. 102985, 2024.

[22] Q. Wang, Y.-Y. Chang, R. Cai, Z. Li, B. Hariharan, A. Holynski, and N. Snavely, "Tracking everything everywhere all at once," *arXiv preprint arXiv:2306.05422*, 2023.

[23] S. Lin, A. J. Miao, J. Lu, S. Yu, Z.-Y. Chiu, F. Richter, and M. C. Yip, "Semantic-super: a semantic-aware surgical perception framework for endoscopic tissue identification, reconstruction, and tracking," in *ICRA*, 2023.

[24] A. Schmidt, O. Mohareri, S. DiMaio, and S. E. Salcudean, "Surgical tattoos in infrared: A dataset for quantifying tissue tracking and mapping," *IEEE Transactions on Medical Imaging*, 2024.

[25] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox, "Flownet 2.0: Evolution of optical flow estimation with deep networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2462–2470.

[26] A. Schmidt, O. Mohareri, S. DiMaio, and S. E. Salcudean, "Fast graph refinement and implicit neural representation for tissue tracking," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 1281–1288.

[27] ——, "Sendd: Sparse efficient neural depth and deformation for tissue tracking," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2023, pp. 238–248.

[28] J. Peng, Q. Chen, L. Kang, H. Jie, and Y. Han, "Autonomous recognition of multiple surgical instruments tips based on arrow obb-yolo network," *IEEE Transactions on Instrumentation and Measurement*, vol. 71, pp. 1–13, 2022.

[29] Z. Zhao, Y. Jin, and P.-A. Heng, "Trasetr: track-to-segment transformer with contrastive query for instance-level instrument segmentation in robotic surgery," in *2022 International conference on robotics and automation (ICRA)*. IEEE, 2022, pp. 11 186–11 193.

[30] N. Ayobi, A. Pérez-Rondón, S. Rodríguez, and P. Arbeláez, "Matis: Masked-attention transformers for surgical instrument segmentation," in *2023 IEEE 20th International Symposium on Biomedical Imaging (ISBI)*. IEEE, 2023, pp. 1–5.

[31] J. Ma, Y. He, F. Li, L. Han, C. You, and B. Wang, "Segment anything in medical images," *Nature Communications*, vol. 15, no. 1, p. 654, 2024.

[32] W. Yue, J. Zhang, K. Hu, Y. Xia, J. Luo, and Z. Wang, "Surgicalsam: Efficient class promptable surgical instrument segmentation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 7, 2024, pp. 6890–6898.

[33] Y. Wang, Y. Long, S. H. Fan, and Q. Dou, "Neural rendering for stereo 3d reconstruction of deformable tissues in robotic surgery," in *International conference on medical image computing and computer-assisted intervention*. Springer, 2022, pp. 431–441.

[34] S. Giannarou, M. Visentini-Scarzanella, and G.-Z. Yang, "Probabilistic tracking of affine-invariant anisotropic regions," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 1, pp. 130–143, 2012.

[35] M. Allan, J. Mcleod, C. Wang, J. C. Rosenthal, Z. Hu, N. Gard, P. Eisert, K. X. Fu, T. Zeffiro, W. Xia, *et al.*, "Stereo correspondence and reconstruction of endoscopic data challenge," *arXiv preprint arXiv:2101.01133*, 2021.

[36] G. Hattab, M. Arnold, L. Strenger, M. Allan, D. Arsentjeva, O. Gold, T. Simpfendörfer, L. Maier-Hein, and S. Speidel, "Kidney edge detection in laparoscopic image data for computer-assisted surgery: Kidney edge detection," *International journal of computer assisted radiology and surgery*, vol. 15, pp. 379–387, 2020.

[37] L. Dinh, J. Sohl-Dickstein, and S. Bengio, "Density estimation using real nvp," *arXiv preprint arXiv:1605.08803*, 2016.

[38] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "Nerf: Representing scenes as neural radiance fields for view synthesis," *Communications of the ACM*, vol. 65, no. 1, pp. 99–106, 2021.

[39] Z. Teed and 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.

[40] J. Sun, Z. Shen, Y. Wang, H. Bao, and X. Zhou, "Loftr: Detector-free local feature matching with transformers," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 8922–8931.