Rethinking causality-driven robot tool segmentation with temporal constraints

Hao Ding1 · Jie Ying Wu2 · Zhaoshuo Li1 · Mathias Unberath1

Purpose Vision-based robot tool segmentation plays a fundamental role in surgical robots perception and downstream tasks. CaRTS, based on a complementary causal model, has shown promising performance in unseen counterfactual surgical environments in the presence of smoke, blood, etc. However, CaRTS requires over 30 iterations of optimization to converge for a single image due to limited observability.

Method To address the above limitations, we take temporal relation into consideration and propose a temporal causal model for robot tool segmentation on video sequences. We design an architecture named Temporally Constrained CaRTS (TC-CaRTS). TC-CaRTS has three novel modules to complement CaRTS—temporal optimization pipeline, kinematics correction network, and spatial-temporal regularization.

Results Experiment results show that TC-CaRTS requires fewer iterations to achieve the same or better performance as CaRTS on different domains. All three modules are proven to be effective.

Conclusion We propose TC-CaRTS, which takes advantage of temporal constraints as additional observability. We show that TC-CaRTS outperforms prior work in the robot tool segmentation task with improved convergence speed on test datasets from different domains.

Keywords Deep learning · Computer vision · Minimally invasive surgery · Computer-assisted surgery · Robustness

Introduction

With the widespread application of surgical robots and the growing demand for autonomous surgery, vision-based robot tool segmentation plays a fundamental role in robot perception [1–10]. In surgical scenes, performances and robustness of segmentation algorithms are both important aspects for the success and safety of downstream operations. Various feed-forward networks and machine-learning techniques designed for semantic/instance segmentation have achieved promising performance [11–18]. However, their performance does not generalize when tested on data from different domains [19].

To improve robustness, some effort has already been made [20–24]. CaRTS [24], designed from a complementary causal model, shows a promising and robust performance when tested on counterfactual surgical environments for robot tool segmentation. However, optimization from an image-wise perspective faces limited observability. This limitation makes CaRTS hard to optimize.

In order to alleviate this issue, we propose a temporal causal model which frames robot tool segmentation along a sequence. This temporal causal model is shown in Fig. 1. We use the same idea from CaRTS where images $I^t$ and segmentation $S^t$ at timestamp $t$ are directly determined by all unobserved robot and camera parameters $T^t$, and all other factors that affect the appearance of the image, noted as the environment $E^t$, at the same timestamp $t$. We assume occlusion has no effect on segmentation and there is no interaction between tools and environments. As the model describes, $T^{t+1}$ at timestamp $t + 1$ are directly determined by $T^t, T^{t-1}$, and all $T$s in the past. Exploring this temporal causal effect might provide temporal constraints that are effective to deal with the issues mentioned above.
We differentiate time-variant and time-invariant factors in T and optimize them separately. Kinematics, with notation $K$, is the representative of time-variant factors. In our experiment, we use joint angles of the patient-side manipulators (PSMs) as the kinematics $K$ and the transformation from the camera frame to the base frames of the PSMs $B$ as the time-invariant factor. We explore the underlining temporal constraints for time-variant factors according to the temporal causal model we proposed. For the optimization of $K_s$, modeling their direct causal effect by $P(K^t \mid K^{t-1}, K^{t-2}, \ldots)$ can be a promising direction since the model can learn motion features from the previous trajectory to provide constraints. Another direction for the optimization of $K_s$ is to explore the spatial-temporal smoothness assumption, which assumes the measurement error and the inter-frame motion is small. This assumption should be safe to make since the difference between two adjacent timestamps is small and the speed of the robot is limited.

Our experiments show that TC-CaRTS effectively reduced the required iterations to achieve the same or better segmentation performance compared to CaRTS. Ablation studies indicate this improvement comes from the temporal constraints utilized by all three proposed modules. The code is available at https://github.com/hding2455/CaRTS.

In summary, the main contributions of this paper are as follows: (1) Proposing temporal causal model for robot tool segmentation in videos and exploring potential direction temporal constraints in this model. (2) Proposing TC-CaRTS that utilizes these temporal constraints to converge with fewer iterations while achieving the same or better performance on robot tool segmentation task compared to its foundation work—CaRTS.

**Related work**

**Robot tool segmentation**

Both image-wise and video-wise semantic/instance segmentation have already been maturely-developed areas. Feed-forward networks, e.g., [11–18], are on a dominating stand. Their variants [1–5, 7] are also the state-of-the-art on robot tool segmentation. At the same time, increasing efforts have been made to incorporate other available information, e.g. geometric information [25–27] and kinematics [8, 9], with visual input to improve performance [6, 8, 9] or robustness [10, 24] for robot tool segmentation.

**Causality in computer vision**

Causality has been receiving increasing attention in computer vision research, especially medical area. Some researchers use ideas from causal inference to design feature representation learning methods [20–23] for domain generalization. Some researchers use the concept of counterfactual for generative models [28, 29]. Lenis et al. [30] use this concept for the interpretability of medical image classifiers. Some researchers focus on posing the underlying causal model of the vision task [24, 31].

**CaRTS**

Ding et al. [24] have proposed a novel causal model where the segmentation is directly determined by the robot kinematics, camera poses, and the environment instead of the observed image. Based on this causal model they design CaRTS architecture that iteratively optimizes feature similarity between rendered images and observed images w.r.t the measured kinematics to estimate true kinematics. The final segmentation is the rendered silhouette of the robot model given the estimated kinematics. CaRTS achieves outstanding robustness across testing domains compared to other feed-forward networks. However, limited observability in image-wise optimization makes CaRTS hard to achieve real-
time inference. Our temporal causal model and TC-CaRTS architecture incorporate temporal constraints that are intuitively helpful for this limitation.

**Method**

We propose the TC-CaRTS architecture based on our temporal causal model. The basic modeling of TC-CaRTS is similar to CaRTS but has three novel modules. We introduce the temporal optimization pipeline first and then introduce KCN and spatial-temporal regularization.

**Temporal optimization pipeline**

The temporal optimization pipeline is illustrated in Fig. 2. Notations are described in the caption of Fig. 2. The optimization objective function in TC-CaRTS is the spatial-temporal regularized ACSLoss between deep feature maps extracted from the observed image $I^t$ and rendered image $\hat{I}^t$. ACSLoss calculates an attentional cosine similarity between feature maps. CaRTS use pre-trained U-Net to extract feature maps from observed image and a hybrid image made up of rendered robot tool and the average background from the training dataset. Visual comparison of the hybrid image and observed image are shown in Fig. 3. Optimizing ACSLoss between these two feature maps aligns the calculated robot configuration to the observed robot. Since the robot tool is differentially rendered, the gradient can be backpropagated to the input kinematics. Thus, gradient descent can be performed to correct the measurement error. More details about the optimization pipeline can be found in [24].

Different from the image-wise CaRTS pipeline, the temporal optimization pipeline differentiates the time-variant factor and time-invariant factor. In TC-CaRTS, time-variant means robot kinematics sequence $K_m^s$ and time-invariant factor means robot base configuration $B$. Other factors $C$ remain constant during optimization. The optimization objective function can be written as Eq. 1 where $\theta$ is KCN’s weights.

$$\arg\min_{\theta, B} ACSLoss(\hat{K}^t, B, C) + R$$

(1)

During optimization, we alternatively perform gradient descent for $\theta$ and $B$. At each timestamp $t$, we first calculate ACSLoss, backpropagate gradient, and perform gradient descent for $\theta$ in KCN for $k - 1$ iterations to learn the temporal relation of kinematics for KCN. At the last iteration, we perform gradient descent for $\theta$ and $B$ jointly. The final corrected kinematics $\hat{K}^t$, base configuration $\hat{B}$, and $C$ are used to render the predicted segmentation $\hat{S}^t$.

**Kinematics correction network**

The Kinematics Correction Network (KCN) is an MLP network $F_\theta$. The weights of the KCN are randomly initialized before each video sequence. The input kinematics $K_m^s$ is a $n \times d$ matrix, $d$ is the dimension of the kinematics, and $n$ is the number of timestamps. $PE(n)$ is the positional encoding for each timestep. The output is the optimized kinematics $\hat{K}^t$ that represents the estimated true kinematics. KCN can be expressed as Eq. 2:

$$\hat{K}^t = K_m^s + F_\theta(K_m^s + PE(n))$$

(2)

**Spatial-temporal regularization**

The spatial-temporal regularization is based on the spatial-temporal smoothness assumptions that (a) the measurement error is small and (b) the inter-frame motion between consecutive timestamps is small. As we use joint angles as kinematics to optimize, we use the L2 norm as regularization function. The Regularization term can be written as:

$$R = \lambda_1 \frac{1}{d} \sum_{i=1}^{d} (K^{t,i} - K^{t-1,i})^2 + \lambda_2 \frac{1}{d} \sum_{i=1}^{d} (K^{t,i} - K^{t-1,i})^2$$

(3)

where $i$ represents the dimension indices of the kinematics $K^{t,i}$ calculated via Eq. 2. $R$ denotes the whole regularization.
term, $\lambda_1, \lambda_2$ are hyperparameters for adjusting regularization strength of each term. $(\hat{K}_{t,i} - K_{t,i})^2$ in the first term regularizes according to assumption (a) and $(\hat{K}_{t,i} - \hat{K}_{t-1,i})^2$ in the second term regularizes according to assumption (b).

**Experiment**

We perform the robot tool segmentation experiment on the dataset from CaRTS [24]. The dataset has nine videos (seven for training, one for validation, and one for testing). Each video contains 400 frames. Each frame has its corresponding kinematics for both patient-side manipulators (PSMs). All videos were recorded under the same camera setting and robot base configuration which are roughly measured before recording. Robot motion recorded in all videos is in free space and no occlusion exists. We train the U-Net [11] feature extractors and all baseline models on the training dataset which only contains videos recorded from one domain without any corruption. We call this domain the regular domain. The validation and test dataset contain counterfactual videos that are recorded on other domains, e.g. smoke, bleeding, etc. Examples of the counterfactual image from different domains are shown in Fig. 4.

In our experiment, we use Dice as the metric for measuring segmentation performance. We first explore the inference speed improvement of our TC-CaRTS architecture compared to CaRTS on the test dataset. We also compare the performance of TC-CaRTS to other deep learning algorithms on the test dataset. Then we present ablation studies on the validation dataset.

**Implementation details**

In our experiment, we use the same implementation setting from CaRTS. For KCN, we choose input length $n = 5$ and optimize all six joint angles and one tool angle of the two PSMs which makes the dimension of the kinematics $d = 14$, KCN has five hidden layers with 32, 64, 128, 128, 64, 32 channels. For spatial-temporal regularization we set $\lambda_1 = 10$ and $\lambda_2 = 1$. We use Adam optimizer with a learning rate of $5 \times 10^{-5} / 3 \times 10^{-6}$ for $\theta / B$. All baselines are trained for 50 epochs with smoke augmentation where smoke is simulated by Fractional Brownian Motion noise. All of the experiments run on a single NVIDIA GeForce RTX 3090 graphic card.

**Speed improvement**

To measure speed improvement, we draw inference time versus performance plots for both CaRTS and TC-CaRTS when inferring with different iteration times $k$ per frame. We choose $k = 1, 2, 3, 5, 10, 30, 50$ and test on the regular domain and bleeding domain. The quantitative results are in Table 1 and the corresponding plot is shown in Fig. 5. From the results, both CaRTS and TC-CaRTS converge at a dice score of 93.6 on the regular domain. TC-CaRTS requires much fewer iterations (5 vs. 50) than CaRTS. On the bleeding domain, TC-CaRTS after a single iteration outperformed CaRTS optimized with 10 iterations. Although introducing KCN increases the inference time for one iteration for TC-CaRTS, reducing the required iteration numbers for each frame results in the improvement of the overall inference speed for TC-CaRTS. Although TC-CaRTS is not real-time yet, it decreases the number of iterations for the optimization to converge. This is a significant step towards real-time inference.

**Performance on robot tool segmentation**

In this experiment, we compare the tool’s Dice score of TC-CaRTS to CaRTS [24], image-based baselines including U-Net [11], HRNet [17], Swin Transformer [16], and method by Colleoni et al [10] and a video-based baseline STCN [18]. All baselines are trained with simulated smoke augmentation. As Table 2 shows, all the feed-forward network-based methods perform well on the regular domain and they can achieve real-time inference. However, their performances deteriorate significantly in other domains that are unseen in the training dataset. TC-CaRTS retains comparable performance as CaRTS on all domains. A qualitative visualization sample of TC-CaRTS’s is shown in Fig. 6.

**Ablation study**

We perform ablation studies on the regular domain of the validation dataset to explore the effectiveness of all modules and design choices.

---

1 https://nullprogram.com/blog/2007/11/20/.
Table 1  Quantitative results of the speed improvement

|          | Regular    | $k = 1$   | 3   | 5   | 10  | 30  | 50  |
|----------|------------|-----------|-----|-----|-----|-----|-----|
| CaRTS    |            | 90.3 ± 3.8| 90.9 ± 3.8| 91.7 ± 3.6| 92.6 ± 3.2| 93.2 ± 3.6| 93.6 ± 2.9|
| Inference time | 213 ms | 345 ms | 526 ms | 961 ms | 2702 ms | 4405 ms |
| TC-CaRTS |            | 91.6 ± 3.9| 93.2 ± 2.8| 93.6 ± 2.7| 93.7 ± 2.7| 93.7 ± 2.8| 3.9 |
| Inference time | 267 ms | 415 ms | 568 ms | 1021 ms | 2712 ms | 4584 ms |
| Bleeding  |            | 90.2 ± 3.8| 90.6 ± 3.7| 91.1 ± 3.7| 91.6 ± 4.0| 91.6 ± 4.0| 91.3 ± 4.5|
| Inference time | 263 ms | 348 ms | 518 ms | 910 ms | 2631 ms | 4347 ms |

Table 2  Robot tool segmentation results

|          | Regular    | Low brightness | Bleeding | Smoke | BG change | FPS |
|----------|------------|----------------|----------|-------|-----------|-----|
| Colleoni’s | 94.9 ± 2.7 | 87.0 ± 4.5 | 55.0 ± 5.7 | 59.7 ± 24.7 | 75.1 ± 3.6 | 35.8 |
| U-Net | 94.9 ± 2.7 | 82.7 ± 3.6 | 52.1 ± 6.8 | 58.8 ± 23.2 | 72.5 ± 3.3 | 37.8 |
| HRNet | 95.2 ± 2.7 | 86.3 ± 3.9 | 56.3 ± 16.4 | 77.2 ± 23.6 | 92.1 ± 4.6 | 15.6 |
| Swin transformer | 95.0 ± 5.5 | 93.0 ± 5.5 | 76.5 ± 9.0 | 82.4 ± 17.0 | 94.8 ± 5.3 | 24.4 |
| STCN | 92.2 ± 2.7 | 64.3 ± 6.9 | 30.8 ± 10.4 | 69.2 ± 26.5 | 84.0 ± 5.6 | 27.5 |
| CaRTS | 93.4 ± 3.0 | 92.4 ± 3.1 | 90.8 ± 4.4 | 91.6 ± 4.7 | 92.3 ± 4.8 | 0.37 |
| TC-CaRTS | 93.6 ± 2.7 | 92.3 ± 3.3 | 92.2 ± 3.3 | 91.9 ± 4.5 | 92.5 ± 3.1 | 1.76 |

Fig. 5  Results of the speed improvement experiment

Fig. 6 Visualization of TC-CaRTS’s robot tool segmentation results on different test domains. White/black: True positives/negatives; Orange/Red: False positives/negatives
Adding KCN can also improve performance when the temporal optimization pipeline can improve performance. But when $k$ becomes larger, it might overfit some frames and fail to generalize without spatial-temporal regularization which is shown as the result when $k = 10$. The spatial-temporal regularization not only improves the performance when $k$ is small and accelerates the convergence but also stabilizes the optimization process of KCN for larger $k$.

### Effectiveness of all modules

We explore the effectiveness of the modules by adding them to the CaRTS architecture. From Table 3, we find that using the temporal optimization pipeline can improve performance. Adding KCN can also improve performance when $k$ is small. However, when $k$ becomes larger, it might overfit some frames and fail to generalize without spatial-temporal regularization which is shown as the result when $k = 10$. The spatial-temporal regularization not only improves the performance when $k$ is small and accelerates the convergence but also stabilizes the optimization process of KCN for larger $k$.

### Input length

We perform an ablation study to see the influence of the input length $n$ for KCN. We test on $n = 1, 3, 5, 10, 40$. From Table 4 we find that if we only use the current frame, i.e. $n = 1$, there will be a limited improvement compared to CaRTS. Once previous frames are provided $n ≥ 3$, the improvement becomes obvious. This indicates that the improvement comes from temporal constraints. However, with a further increase in the input length, the performance does not increase. We suppose this is because increasing the input length will also increase the difficulty of the optimization.

### Regularization strength

We perform an ablation study to see the influence of the regularization strength $\lambda_1, \lambda_2$ in the spatial-temporal regularization. We separately test on $\lambda_1 = 0, 1, 10, 100, 1000, 10,000$ when $\lambda_2 = 1$ and $\lambda_2 = 0, 0.1, 1, 10, 100, 1000$ when $\lambda_1 = 10$. From Table 5 we find that when we set either $\lambda_1$ or $\lambda_2$ to 0, there are performance drops (1.5/0.5 dice score) compared to the default setting. When regularization is too strong ($\lambda_1 ≥ 1000, \lambda_2 ≥ 100$), optimization might also become harder.

### Effectiveness over time

We perform an ablation study to show an insight into the TC-CaRTS’s effectiveness over time. We calculate the average Dice difference of the first $m$ frames for $m = 1, 2, \ldots, 400$ between TC-CaRTS and CaRTS. The result plots are shown in Fig. 7. As the plots show, the Dice difference increases as $m$ increases. This indicates that with more temporal information having been processed, TC-CaRTS’s advantage with temporal constraints becomes more obvious. We also find that when $m$ is small (≤ 50), TC-CaRTS performs worse than CaRTS when $k = 5$. This indicates that TC-CaRTS might overfit some early frames. However, when more $m$ increases, TC-CaRTS start to outperform CaRTS.

### Reducing iteration number over time

Inspired by the above result, we design an ablation study to compare the performance for fixed $k = 10$ and gradually reduced $k = ⌈\frac{10}{(n+1)/1000}\rceil$ value. The results from Table 6 indicate that gradually reducing the iteration number doesn’t harm the performance of the model and can improve the inference speed by an impressive margin (1.81 vs. 1.09 FPS). This further indicates that a better optimization scheme can be explored to further improve the inference speed of TC-CaRTS.
Table 6 Test results for decreasing iteration numbers $k$ during optimization

| $k = \left\lceil \frac{10}{(n+1)/100} \right\rceil$ | Regular | Low brightness | Bleeding | Smoke | BG change | FPS |
|---|---|---|---|---|---|---|
| 93.3 ± 2.5 | 91.8 ± 2.5 | 90.7 ± 2.5 | 91.3 ± 2.5 | 92.8 ± 2.3 | 1.8 |
| $k = 10$ | 93.3 ± 2.5 | 91.9 ± 2.7 | 90.7 ± 3.6 | 91.7 ± 3.8 | 92.9 ± 2.6 | 1.1 |

**Limitations**

Although TC-CaRTS is proven to be effective, there are still limitations that are not fully resolved. Firstly, the challenge to achieve real-time remains. On the one hand, differentiable rendering and backpropagation require more computation than a single feed-forward network. On the other hand, redundant feature extraction operations on similar rendered images also restrict the inference speed. What’s more, better options for the network design and optimization scheme may exist that can further reduce the required iterations with more informative temporal constraints. Secondly, the architecture works under the assumption that there is no occlusion and interaction. To deal with occlusion and interaction, information for the environmental factor $E$ is necessary. The representation of $E$ might be estimated through vision or other sensors. All of these limitations also imply essential directions for future work.

**Conclusion**

In summary, limited observability causes slow convergence for CaRTS. We propose a temporal causal model and explore underlying temporal constraints in this model. Inspired by the temporal causal model, we propose TC-CaRTS with three novel modules to complement CaRTS– temporal optimization pipeline, kinematics correction network, and spatial-temporal regularization. TC-CaRTS requires fewer iterations to achieve the same or better performance as CaRTS while achieving the same or better performance in different domains compared to CaRTS. Ablation studies indicate that all modules are effective and the effectiveness comes from temporal constraints.

**Acknowledgements** This research is supported by a collaborative research agreement with the MultiScale Medical Robotics Center at The Chinese University of Hong Kong.

**Funding** Hao Ding is funded by the MultiScale Medical Robotics Center at The Chinese University of Hong Kong. This article does not contain any studies with human participants or animals performed by any of the authors. This article does not contain patient data.

**References**

1. García-Peraza-Herrera LC, Li W, Fidon L, Grujthuijzen C, Devreker A, Attilakos G, Depret J, Poorten EBV, Stoyanov D, Vercauteren T, Ourselin S (2017) ToolNet: holistically-nested real-time segmentation of robotic surgical tools. In: Proceedings of the IROS.
2. Jin Y, Cheng K, Dou Q, Heng P (2019) Incorporating temporal prior from motion flow for instrument segmentation in minimally invasive surgery video. In: Proceedings of the MICCAI.
3. Shvets AA, Rakhlin A, Kalinin AA, Iglovikov VI (2018) Automatic instrument segmentation in robot-assisted surgery using deep learning. In: Proceedings of the ICMLA.
4. Pakhomov D, Premachandran V, Allan M, Azizian M, Navab N (2019) Deep residual learning for instrument segmentation in robotic surgery. In: Proceedings of the MLMI
5. Islam M, Atputharuban DA, Ramesh R, Ren H (2019) Real-time instrument segmentation in robotic surgery using auxiliary supervised deep adversarial learning. IEEE Robot Autom Lett 4:2188
6. Qin F (2019) Surgical instrument segmentation for endoscopic vision with data fusion of CNN prediction and kinematic pose. In: Proceedings of the ICRA
7. Zhao Z, Jin Y, Lu B, Ng C, Dou Q, Liu Y, Heng P (2021) One to many: adaptive instrument segmentation via meta learning and dynamic online adaptation in robotic surgical video. In: Proceedings of the ICRA
8. Su Y-H, Huang K, Hannaford B (2018) Real-time vision-based surgical tool segmentation with robot kinematics prior. In: 2018 international symposium on medical robotics (ISMR). IEEE, pp 1–6
9. da Costa Rocha C, Padoy N, Rosa B (2019) Self-supervised surgical tool segmentation using kinematic information. In: 2019 international conference on robotics and automation (ICRA). IEEE, pp. 8720–8726
10. Colleoni E, Edwards PJ, Stoyanov D (2020) Synthetic and real inputs for tool segmentation in robotic surgery. In: Proceedings of the MICCAI (2020)
11. Ronneberger O, Fischer P, Brox T (2015) U-Net: convolutional networks for biomedical image segmentation. In: Proceedings of the MICCAI
12. Chen L, Zhu Y, Papandreou G, Schroff F, Adam H (2018) Encoder-decoder with atrous separable convolution for semantic image segmentation. In: Proceedings of the ECCV
13. He K, Gkioxari G, Dollár P, Girshick RB (2017) Mask R-CNN. In: Proceedings of the ICCV
14. Chen K, Pang J, Wang J, Xiong Y, Li X, Sun S, Feng W, Liu Z, Shi J, Ouyang W, Loy CC, Lin D (2019) Hybrid task cascade for instance segmentation. In: Proceedings of the CVPR
15. Ding H, Qiao S, Yuille AL, Shen W (2021) Deeply shape-guided cascade for instance segmentation. In: Proceedings of the CVPR
16. Liu Z, Lin Y, Cao Y, Hu H, Wei Y, Zhang Z, Lin S, Guo B (2021) Swin transformer: hierarchical vision transformer using shifted windows. In: Proceedings of the ICCV
17. Wang J, Sun K, Cheng T, Jiang B, Deng C, Zhao Y, Liu D, Mu Y, Tan M, Wang X, Liu W, Xiao B (2019) Deep high-resolution representation learning for visual recognition. In: TPAMI
18. Cheng HK, Tai Y-W, Tang C-K (2021) Rethinking space-time networks with improved memory coverage for efficient video object segmentation. In: NeurIPS
19. Drenkow N, Sani N, Shpitser I, Unberath M (2021) Robustness in deep learning for computer vision: mind the gap? arXiv:2112.00639
20. Mitrovic J, McWilliams B, Walker JC, Buesing LH, Blundell C (2021) Representation learning via invariant causal mechanisms. In: Proceedings of the ICLR
21. Ouyang C, Chen C, Li S, Li Z, Qin C, Bai W, Rueckert D (2021) Causality-inspired single-source domain generalization for medical image segmentation. arXiv:2111.12525
22. Zhang C, Zhang K, Li Y (2020) A causal view on robustness of neural networks. In: Larochelle H, Ranzato M, Hadsell R, Balcan M, Lin H (eds) Proceedings of the NIPS
23. Liu C, Sun X, Wang J, Tang H, Li T, Qin T, Chen W, Liu T-Y (2021) Learning causal semantic representation for out-of-distribution prediction. In: Proceedings of the NIPS
24. Ding H, Zhang J, Kazanzides P, Wu JY, Unberath M (2022) Carts: causality-driven robot tool segmentation from vision and kinematics data. In: Proceedings of the MICCAI. Springer, pp. 387–398
25. Allan M, Ourselin S, Hawkes DJ, Kelly JD, Stoyanov D (2018) 3D pose estimation of articulated instruments in robotic minimally invasive surgery. IEEE Trans Med Imaging 37(5):1204–1213
26. Li Z, Liu X, Drenkow N, Ding AS, Creightton FX, Taylor RH, Unberath M (2021) Revisiting stereo depth estimation from a sequence-to-sequence perspective with transformers. In: Proceedings of the ICCV
27. Ye M, Zhang L, Giannarou S, Yang G (2016) Real-time 3d tracking of articulated tools for robotic surgery. In: Proceedings of the MICCAI
28. Reinhold JC, Carass A, Prince JL (2021) A structural causal model for MRI images of multiple sclerosis. In: Proceedings of the MICCAI
29. Pawlowski N, de Castro DC, Glocker B (2020) Deep structural causal models for tractable counterfactual inference. In: Proceedings of the NIPS
30. Lenis D, Major D, Wimmer M, Berg A, Sluiter G, Bühler K (2020) Domain aware medical image classifier interpretation by counterfactual impact analysis. In: Proceedings of MICCAI
31. Castro DC, Walker I, Glocker B (2020) Causality matters in medical imaging. Nat Commun 11(1):3673

Publisher’s Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.