Proceedings of 2022 IEEE
International Conference on Mechatronics and Automation
August 7 - 10, Guilin, China

Evaluation of an Autonomous Navigation Method for Vascular Interventional Surgery in Virtual Environment

Fanxu Meng [1], Shuxiang Guo [1*], Wei Zhou [1], Zhengyang Chen [1]

[1] *Key Laboratory of Convergence Biomedical Engineering System and Healthcare Technology, The Ministry of Industry and Information Technology, School of Life Science, Beijing Institute of Technology, No.5, Zhongguancun South Street, Haidian District, Beijing 100081, China*

E-mails: {mengfanxu & guoshuxiang & zhouwei & chenzhengyang}@bit.edu.cn;

* Corresponding author

***Abstract* - With the increasing use of vascular interventions, catheter navigation in complex vessels has become even more critical. Vascular intervention surgeries also require more precise manipulation and a high level of automation. In this paper, a reinforcement learning- based navigation model was designed and implemented in virtual environment. The task is to insert the catheter and guidewire into the aortic arch automatically. The whole experiment was carried out in the virtual environment based on SOFA engine. Firstly, the model of vessel and catheter/guidewire was established. Secondly, a reinforcement learning method (asynchronous advantage actor-critic) was used to test the performance of catheter autonomous navigation in vessels. Finally, the results were analyzed and compared with manual operation. The results demonstrate that the automatic insertion has shorter operation time and less contact force.**

Index Terms -Vascular interventional surgery, Reinforcement learning, Catheter insertion, Actor-critic.

I. INTRODUCTION

With the rapid increase in cardiovascular morbidity, minimally invasive vascular interventions have rapidly replaced traditional open or cranial surgery due to their patient-friendly nature [1]. For many years, medical robots have been used in surgery and healthcare, and the use of robots in surgery has been beneficial for departments such as head and neck, cardiac and urology Robotics in cardiac surgery [2].

The ultimate goal of vascular intervention is to perform the procedure without compromising the integrity of the chest [3]. Catheter access to the heart can be less traumatic for the patient. However, it significantly increases the complexity of the surgical approach, requiring more sophisticated instruments, greater precision, dexterity and intuitive remote manipulation [4]. It also has its disadvantages: interventional procedures are highly dependent on the surgeon's surgical experience, and the cost of training qualified surgeons is high [5].

Existing master-slave interventional robot systems are passive recipients of the surgeon's actions. In the more mature CorPath GRX system, autonomy is also limited to compensation for the angle of rotation of the guidewire [6]. It is necessary to combine simulation technology and in vitro physical vascular models to build a more intelligent robotic system for vascular interventions.

There is also growing research on virtual surgery. The essential features of virtual surgical systems are the realism of the model and real-time interaction. The most challenging task in developing virtual training systems is how to accurately and efficiently simulate the behaviour of interventional devices (e.g. guidewires and catheters) in the vascular system. Although many approaches have been proposed in this area, most of them have focused on the behaviour of individual devices [7]. However, the human circulatory system has a high degree of morphological complexity. In endovascular procedures, there are many vascular bifurcations from the incision to the target location [8]. To guide the interventional device to the desired branched vessel, the surgeon needs to translate and rotate the distal end of the device to control the movement of its tip to select the correct path. This task is difficult to accomplish with only one device. In actual clinical practice, the surgeon needs to operate multiple interventional devices (adjustable sheaths, catheters, catheters, guidewires, etc.) simultaneously to access the target location [9]. In order to closely resemble a real surgical scenario as much as possible, the virtual training system must be able to simulate the interaction behaviour of multiple devices.

In recent years, automation technologies based on deep learning and reinforcement learning have been rapidly shaped and implemented, showing a high scientific value. In the field of surgical robots, the automated and intelligent operation of surgical instruments based on ensuring surgical safety has been a goal pursued by researchers. However, compared to general artificial intelligence systems, surgical data is costly to acquire and less interpretable, and a model is only valid for a single surgical task and lacks generalization capability [10]. However, in contrast to the complex multi-process decision-making tasks such as resection and suturing in general surgery, the simple, intravascular dynamic and stable mode of operation of tube-filament access in interventional surgery provides a good environment for the training of automatic models and makes it easier to exploit the advantages of high precision and stability of robotic control. The study of critical technologies for the automatic control of surgical robotic tube and wire access is a realistic and scientific attempt [11]. The simplification and automation of tube and wire access operations can greatly relieve the technical and experiential pressure on interventional surgeons and provide new perspectives and tools for the development of surgical robots and vascular surgery by quantifying the implicit intraoperative features.

Researchers from different teams have experimented with virtual training systems, automated catheter navigation in vascular in vitro models. Yang et al. at Imperial College of Technology attempted to use reinforcement learning algorithms to control a vascular interventional robot to complete autonomous over-arch operations on four aortic arch models, showing that the robot's operating force fluctuated over a significantly lower range than that of a human hand and operated at approximately half the speed of a human hand [12]. Using the dueling deep Q-learning (DQN) algorithm to control catheter entry into a heart model, You et al. at the University of Ulsan also demonstrated that a reinforcement learning strategy based on a simulated environment could control an actual catheter to complete a cardiac entry [13]. However, there are still problems such as lack of accuracy and a relatively simple model. Initial control using the Deep Deterministic Policy Gradients (DDPG) algorithm was implemented in a 2D environment by Karstensen et al. at Fraunhofer IPA, Germany, and performed well in a planar vascular model, but fell short for higher-level path control [14].

Most studies have been devoted to simulating the behaviour of a single individual and blood vessels in the human vasculature, with fewer studies on interaction, but in real surgical operations, the surgeon needs to work with multiple instruments in order to complete the entire surgical operation [15]. So, it is vital to regard both the catheter and guidewire in vascular interventions.

In this paper, we use a reinforcement learning algorithm to implement control of catheter access in a simulation engine. The navigation process was tested to be effective and compared with the manual manipulation. The simulation process performed is an over-arching operation of the aortic arch, which is eventually trained successfully in a virtual environment. Meanwhile, manual manipulation was also conducted using phantoms as the input of the motion information. The paper is structured as follows: Section II focuses on the building of vascular models and reinforcement learning methods. Section III introduces the arrangements of the experiment and analyse the results. Section IV and V present the discussion and the summarization of the paper.

II. METHODS

The methods we use in this research includes four main parts: modelling, reinforcement learning, the specific algorithm we use and the training engine. The mentioned four parts cover the two main issues- algorithms and simulation environments for reinforcement learning.

A. *Modeling*

The process of modelling the aortic arch and catheter progresses from the base shape to the inclusion of more features. Its medical characteristics are primarily realistic, and the established model is subsequently fed into the SOFA engine to establish its environmental parameters.

In clinical vascular interventions, the proximal end of the guidewire and catheter is controlled by the surgeon, while its distal end moves within the bounded range of the vascular constraint in response to the proximal surgeon's maneuver [16]. In the simulation modeling of the guidewire catheter, in order to reduce the computational volume and improve the simulation effect, the guidewire catheter is often discretized (i.e., the longer guidewire catheter is decomposed into a limited number of small, interrelated segments), so that from the effect, each segment of the guidewire catheter can be considered as a discrete cantilever beam structure. Therefore, this section will propose a physical model construction method of the catheter guidewire using Timoshenko beam theory based on the dynamics of the cantilever beam, which can be achieved by imposing different physical parameters to model the guidewire and catheter with different physical properties such as stiffness and mass, respectively.

The movement of the guidewire through the vascular tree is simulated using SOFA. The walls of the vascular tree are rigid. The lumen is empty; thus, no dynamic resistance to the guidewire motion is considered.

Our model is based on the angiographic image of the aortic arch, but with a partial simplification of the vascular connection at the vessel cross-section, so that the cross-sections of the model vessels are all circular, while the interface is a smooth connection.

In order to verify the function of reinforcement learning, this paper obtains the accuracy and stability of catheter autonomy learning by modeling simulated over-arch operations on the aortic arch.

B. *Reinforcement Learning Methods*

The basic model of reinforcement learning is the individual-environment interaction. The individual/intelligent agent is the part of the individual that can take a series of actions and expects to achieve a high benefit or goal. The other parts associated with this are referred to as the environment. The whole process is discretized into different time steps. At each moment, the environment and the individual interact accordingly. The individual can take specific actions, which are imposed on the environment. After receiving the individual's action, the environment gives the individual feedback on the current state of the environment and on the reward that has been generated as a result of the previous action.

Reinforcement learning is a formal framework that uses Markov decision processes to define the process by which a learning intelligence interacts with its environment using states, actions and gains [17].

In the basic setup of reinforcement learning, there are essential elements such as agent, environment, action, state, reward, etc. The agent interacts with the environment to generate trajectories, and by performing the action, the environment changes its state. The agent interacts with the environment, generating trajectories that cause the environment to change state by performing an action; the environment then gives the agent a reward (positive or negative) for its current action. Through this interaction, more experience is accumulated, and the policy is updated to finally form a closed loop. The mystery of why reinforcement learning can model the long-term benefits of decision-making lies in its optimization goal. To be specific, at each moment, the reward is a specific value and the agent's goal is to maximize the expectation of the reward it obtains [18]. This means that instead of maximizing the immediate reward, it maximizes the cumulative reward over time.

As shown in Fig. 1, the guidewire and catheter are both set as the agents and could move forward/backward as well as rotate. The algorithm is able to control them separately. In the figure, q1 and q2 are the two nodes of the beam unit e1, each with 6 degrees of freedom. Again, because the overall morphology of the guidewire changes considerably in the vasculature, but maintains a small deformation in the local coordinate system of each beam cell, in accordance with the assumption of the co-rotation model.

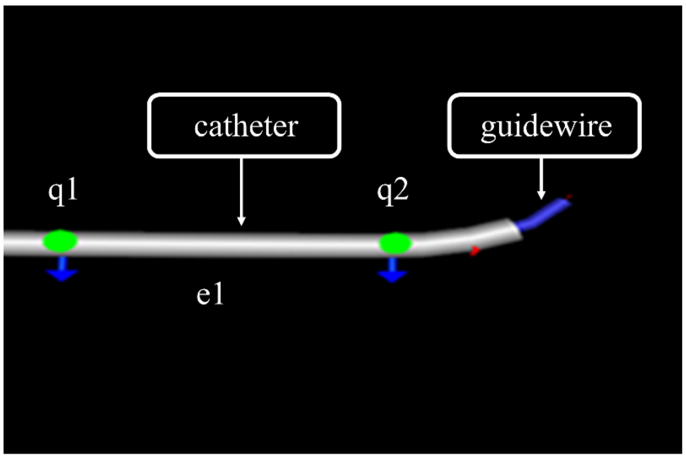


Fig. 1. Agent setting of both guidewire and catheter.

C. *Asynchronous Advantage Actor-critic(A3C)*

In this section, the background of the emergence of A3C and its advantages are presented with a contrast between previous methods. Finally, the single A3C network algorithm is given.

Asynchronous advantage actor-critic is an algorithm proposed by Google DeepMind to solve the Actor-Critic non-convergence problem. While DQN is vital because it has an experienced pool that reduces the correlation between data, A3C proposes an alternative way to reduce the correlation between data: asynchronously.

A3C creates multiple parallel environments and allows multiple agents with sub-structures to update parameters in the main structure on these parallel environments simultaneously. The agents in parallel do not interfere with each other, while the parameter updates of the primary structure are interfered with by the discontinuity of the updates submitted by the substructures, so the correlation of the updates is reduced, and convergence is improved.

The main idea of A3C is asynchronous, corresponding to the asynchronous distributed RL framework. Corresponding to Google's Gorilla platform Massively Parallel Methods for Deep Reinforcement Learning in 2015, Gorilla uses different machines with the same PS [19]. While in A3C, it is the same machine with multi-core CPUs, which reduces the parameter, and in A3C, it is the same machine with multiple CPUs, which reduces the cost of transferring parameters and gradients, and the validation iterations are significantly faster in the paper. And more importantly, it is an actor-learner pair with multiple threads on the same machine; each thread corresponds to a different exploration policy, and the overall inter-sample correlation is low, so it is no longer necessary to introduce an experience replay mechanism in DQN for training. This enables an on-policy approach to training. In addition, the CPU is used in training instead of the GPU because the RL batch is generally small during training and the GPU is much idle while waiting for new data.

Different types of deep neural networks provide an efficient operational representation of the policy optimization task in DRL. To alleviate the instability that occurs when combining traditional policy gradient methods with neural networks, various types of deep policy gradient methods use an empirical replay mechanism to eliminate the correlation between training data.

However, there are two main problems with the empirical replay mechanism: Each real-time interaction between the agent and the environment requires a lot of memory and computational power; the experience replay mechanism requires the agent to learn using an off-policy approach, which can only be updated based on the data generated by the old policy; and the training of DRLs has previously relied on computationally powerful graphics processors.

The A3C algorithm first constructs a global network. This network will consist of convolutional layers for spatial dependencies, followed by LSTM layers for temporal dependencies, and finally value and policy output layers [20]. The process of the algorithm is shown below:

Algorithm 1. A3C network learning process

1 **Input**: public part of the A3C neural network parameters θ, ω
2 Update time series t=1
3 Reset the gradient updates of Actor and Critic
4 $\theta'=\theta$, $\omega'=\omega$
5 Initialize state start t
6 Choose action a_t based on policy $\pi(a_t| s_t; \theta)$
7 Execute action a_t to get reward r_t and new state
8 $t\leftarrow t+1, T\leftarrow T+1$
9 If s_t is terminated, then go to step 10, otherwise go back to step 6
10 Calculate Q(s,t) for the last time series position s_t
11 For $i \in (t-1, t-2, \ldots .t_{start})$:
  1) Calculate Q(s, i) for each moment:
  $Q(s,i)=r_i+\gamma Q(s,i+1)$
  2) Local gradient update of the cumulative Actor
  3) Local gradient update of the cumulative Critic
12 Update the model parameters of the global neural network.
  $\theta=\theta-\alpha d\theta$, $w=w-\beta d\omega$
13 If $T>T_{max}$, then the algorithm ends and **outputs** the public part of the A3C neural network parameters θ, ω, otherwise go to step 3

D. *Training Engine*

The setup consists of the DRL-Agent, a simulated environment created in SOFA.

SOFA (Simulation Open Framework Architecture) is an open-source C++ library originally developed for interactive computational medical simulation [21]. SOFA, as a physical simulation engine mostly used for medical simulation scenario studies, has the following advantages:

1)SOFA decomposes the complex simulator into independent components with different functions, such as degrees of freedom, collision detection algorithm, binding force, model properties, differential equation solver, linear solver, etc. 2) Users can call the required components in the same scenario or data structure according to the simulation requirements, and combine them by layer and adopt the same mapping mechanism to build personalized and customized simulation scenarios to achieve real-time synchronization and continuity of the simulation process.

SOFA physical engine introduces the concept of multi-model representation based on scenario graphs to easily build simulations consisting of any number of objects. The physical object model constructed in SOFA engine is a unified body of internal model, collision model and visual model, where: the internal model has independent degrees of freedom, mass law and intrinsic law, which can be used to calculate the elastic force of the model, etc.; the collision model is in contact with the internal geometry model, which can be used to realize collision detection and response; the visual model contains detailed geometry and rendering parameters, which can be used to perform simulation The visual model contains detailed geometry and rendering parameters for visual rendering of the object. The inclusion of the collision model and the visual model can avoid the problem of low real time and realism caused by the physical model calculation. In the creation of personalized simulation environments, the model construction and mapping mechanisms are the first task to be studied.

Most of the current games have a large number of SOFA games, a perfect engine, and a good training environment to build. Since SOFA can be cross-platform, it can be trained under Windows and Linux platforms and then converted to WebGL for publishing to the web [22]. Furthermore, ml-agents is an open-source plug-in for SOFA, which allows developers to train in SOFA's environment, without even writing code in python, without a deep understanding of PPO, SAC and other algorithms. As long as developers configure the parameters, they can easily use reinforcement learning algorithms to train their own models [23].

During the training process, the user's operating information is collected by the force interaction device and input into the computer to control the movement of the surgical instruments in the virtual training environment. The interaction between the surgical instruments and the vascular tissue will be performed in the computer, and the simulation results will be transmitted to the force interaction device and the LED display, and finally fed back to the user as tactile force and visual images, respectively. The experimental setting of the manipulator and the virtual engine is shown in Fig. 2. We use a personal laptop to build the virtual simulation environment for our vascular interventional surgery training simulator. The laptop used has 16 GB of RAM, Intel Core i7-10750H processor and NVDIA RTX2070 graphic card.

Fig. 2. Experimental setting of the manipulator and virtual engine.

III. Experiments and Results

The experiment was carried out both in virtual environment and the real control. As the experimental setting shown in Fig. 2, we conduct the experiment in the virtual engine using 2 set of phantoms. The phantoms could control the catheter and the guidewire in the same time or separately, guiding them into the aortic arch. Besides, the training process could also be conducted in that particular virtual environment. In the process of the experiment, the conduct force between the vessel and the catheter is accumulated, and experiment time is also recorded. The learning process in the virtual environment is shown as Fig.4.

The results of A3C learning and manual operation are shown in Fig. 4. As shown in the figure, the final results obtained from the training can corroborate that the model was successfully trained to achieve stable returns, with large fluctuations in the choice of losses, which may be related to the instability of the model itself at the time of collision.

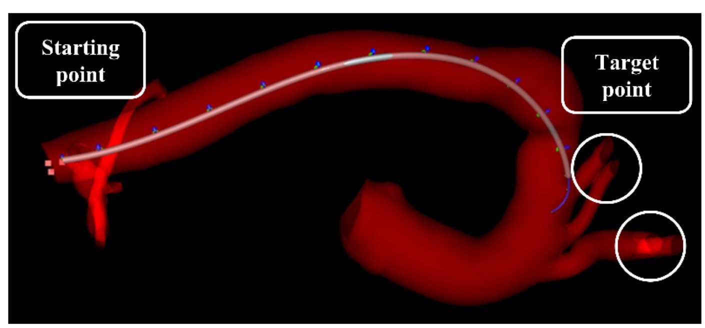
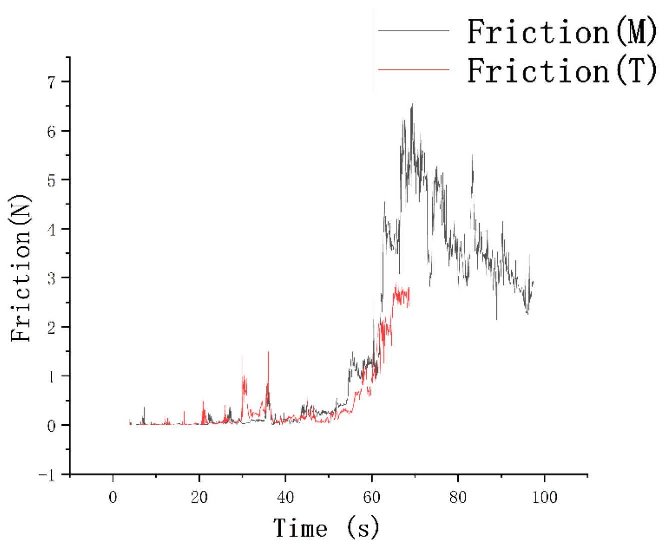
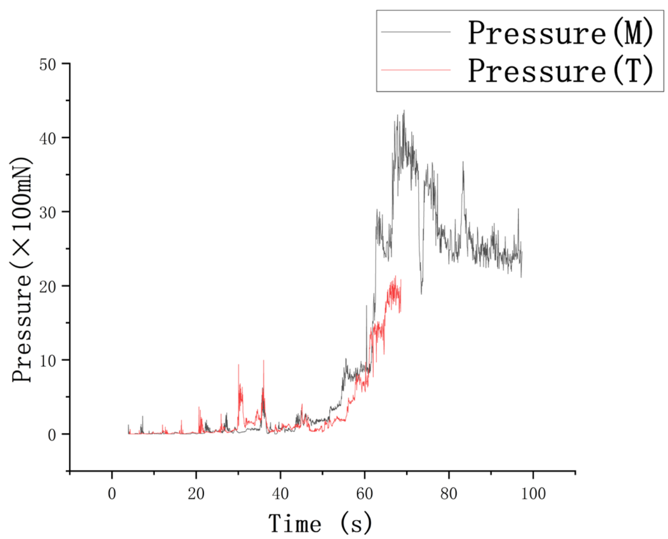


Fig. 3. RL learning process of catheter in SOFA.

The A3C algorithm was added and retrained for comparison to obtain different results of the algorithm for the simulation of vascular interventional procedures in the built training environment. The virtual environment for training is represented in Fig. 3. And using the catheter to access the aortic arch and perform over-arch manipulation, we derived the subsequent result as shown in Fig. 4.



(a)



(b)

Fig.4. Results of contact force in both learning setting and manual experiment: (a) Result of the friction. (b) Result of the pressure.

The conclusion that can be drawn from Fig. 4 is that after training using the reinforcement learning method, the catheter and guidewire can enter the aortic arch in a shorter time. And the contact force (friction, pressure) between the catheter and the vessel in the simulation environment is reduced more significantly, as shown in TABLE I.

The results show that the average insertion time after the training is 68.61s, which is nearly 30% shorter than manual insertion. Also, the average contact force as well as the maximum force of the training also perform better than the manual manipulation. But there are still forceful fluctuations during the entry, which shows that the model needs further optimization. Besides, our experiment needs to include more parameters.

TABLE I
RESULTS OF THE TRAINING PROCESS AND MANUAL MANIPULATION

	Training	Manual
Insertion Time (s)	68.61	97.35
Contact Force (max) (mN)	22.15	43.27

IV. DISCUSSIONS

The research in this paper is divided into two main processes: the modelling part and the model training part.

In the modelling part, we try to provide an environment for subsequent algorithm training by modelling the aortic arch. The model of the aortic arch is based on the real vascular environment, but due to the high complexity of the vasculature and the difficulty to monitor the internal environment in real time, the modelling process simplifies the structure of the vasculature and ignores the blood flow and the more complex respiration and pulsation in the vasculature, which needs to be solved by more accurate modelling and imposing fluid motion in the model.

In the training part of the model, the final training results obtained using the A3C algorithm showed that the reward training of the model was more in line with expectations and was able to achieve reward stability in a short step; however, the fluctuation of the loss function was large and did not converge after a certain length of training, which may be due to the parameter settings in the training, or the large fluctuation of the catheter position in the training environment This may be due to the parameter settings in the training, or it may be due to the large fluctuations in the position of the conduit in the training environment, which does not converge to the same stable path. These problems need to be solved by tuning the parameters and imposing more constraints on the catheter in subsequent studies.

V. CONCLUSIONS

In this paper, we aim to use reinforcement learning algorithms to implement control of catheter access in a simulation engine. The simulation process performed is an over-arching operation of the aortic arch, which is eventually trained successfully in a virtual environment and can be used in subsequent catheter access navigation in a real environment. As the result shows, this concept is feasible and can further improve the model accuracy and algorithm accuracy.

To date, we have been investigating the feasibility of our approach using highly idealized vascular models and just a guidewire. But there is still a long way to go before reinforcement learning can be applied to real-world scenarios. The complex environment that changes at any time in the vasculature requires a simulation environment with sufficient vascular complexity, while being able to simulate blood flow, pulse, heartbeat and other influencing factors, which will be a huge project. The results show that the A3C algorithm is able to obtain more desirable results in these idealized models. Future studies should address the mentioned limitations by adjusting the agents and settings to fit more realistic vessel geometries, as well as by using both guidewires and catheters. Also, the specific contact of the vessel wall with the catheter guidewire and its own elastic characteristics should also be properly characterized.

ACKNOWLEDGEMENT

This work was supported in part by the National Natural Science Foundation of China (61703305), in part by National High-tech Research and Development Program (863 Program) of China (2015AA043202).

REFERENCES

[1] S. Guo, et al, "Machine learning-based operation skills assessment with vascular difficulty index for vascular intervention surgery," *Medical & Biological Engineering & Computing*, vol. 58, no. 8, pp. 1707- 1721, 2020.

[2] Swaminathan R V, Rao S V, "Robotic-assisted transracial diagnostic coronary angiography," *Catheterization & Cardiovascular Interventions*, vol. 10, no. 1, pp. 7-21, 2018.

[3] Y. Zhao, et al, "A novel noncontact detection method of surgeon's operation for a master-slave endovascular surgery robot," *Medical & Biological Engineering & Computing*, vol. 58, no. 4, pp. 871-885,2020.

[4] Y. Wang, et al, "Online measuring and evaluation of guidewire inserting resistance for robotic interventional surgery systems," *Microsystem Technologies*, vol. 24, no. 8, pp. 3467-3477, 2018.

[5] Britz. G, et al, "Neuro-endovascular-specific engineering modifications to the CorPath GRX robotic system," *Journal of neurosurgery*, vol. 1, no. 1, pp. 1-7, 2019.

[6] C. Yang, et al, "A vascular interventional surgical robot based on surgeon's operating skills," *Medical & Biological Engineering & Computing*, vol. 57, no. 9, pp. 1999-2010, 2019.

[7] J. Guo, et al, "A vascular Interventional surgical robotic system based on force-visual feedback," *IEEE Sensors Journal*, vol.19, no. 23, pp. 11081-11089, 2019.

[8] Chi. W, et al, "Trajectory optimization of robot-assisted endovascular catheterization with reinforcement learning," *IEEE International Conference on Intelligent Robots and Systems,* pp. 3875-3881,2018.

[9] Hyeonseok, et al. "Automatic control of cardiac ablation catheter with deep reinforcement learning method," *Journal of Mechanical Science and Technology*, vol. 33, no. 11, pp. 5415-5423, 2019.

[10] Karstensen. L, et al. "Autonomous guidewire navigation in a two-dimensional vascular phantom," *Current Directions in Biomedical Engineering*, vol. 6, no. 1, 2020.

[11] K. Wang, et al, "Endovascular intervention robot with multi-manipulators for surgical procedures: Dexterity, adaptability, and practicability," *Robotics and Computer-Integrated Manufacturing*, vol. 56, pp. 75-84, 2019.

[12] J. Kim, et al, "A novel tip-positioning control of a magnetically steerable guidewire in sharply curved blood vessel for percutaneous coronary intervention," *International Journal of Control, Automation and Systems*, vol. 17, no. 8, pp. 2069-2082, 2019.

[13] Y. Zhao, et al, "A CNNs-based prototype method of unstructured surgical state perception and navigation for an endovascular surgery robot," *Medical & Biological Engineering & Computing*, vol. 57, no. 9, pp. 1875-1887, 2019.

[14] Khan E M, et al, "First experience with a novel robotic remote catheter system: Amigo™ mapping trial," *The Journal of Interventional Cardiac Electrophysiology*, vol.37, no.2, pp.121-129, 2013.

[15] X. Yin, et al, "Safety operation consciousness realization of MR fluids-base novel haptic interface for teleoperated catheter minimally invasive neuro surgery," *IEEE/ASME Transactions on Mechatronics,* vol. 21, no. 2, pp. 1043-1054, 2015.

[16] W. Zhou, et al, "Kinetics analysis & ADRC-based controller for a String-driven Vascular Intervention Surgical Robotic System", *Micromachines*, Vol.13, No.770. https://doi.org/10.3390/mi13050770, 2022.

[17] S. Guo, et al, "A novel robot-assisted endovascular catheterization system with haptic force feedback," *The IEEE Transactions on Robotics*, vol. 35, no. 3, pp. 685-696, 2019.

[18] Y. Wang, et al, "Surgeons operation skill-based control strategy and preliminary evaluation for a vascular interventional surgical robot," *Journal of Medical and Biological Engineering*, vol. 39, no. 5, pp. 653-664, 2019.

[19] Y. Zhao, et al, "A CNNs-based prototype method of unstructured Surgical state perception and navigation for an endovascular surgery robot", *Medical & Biological Engineering & Computing*, vol.57, no.9, pp.1875-1887, DOI:10.1007/s11517-019-02002-0, 2019.

[20] W. Zhou, et al, "A Surgeon`s Habits-Based Novel Master Manipulator for the Vascular Interventional Surgical Master-Slave Robotic System", *IEEE Sensors Journal*, Vol.22, No.10，pp.9922-9931, 2022..

[21] Y. Zhao, et al, "Operating force information on-line acquisition of a novel slave manipulator for vascular interventional surgery," *Biomed Microdevices*, vol. 20, no. 2, pp. 1-13, 2018.

[22] C. Yang, et al, "Cloud Communication-based Sensing Performance Evaluation of a Vascular Interventional Robot System", *IEEE Sensors Journal*, Vol.22, No.99, pp. 9005-9017, 2022.

[23] X. Jin, et al, "Development of a tactile sensing robot-assisted system for vascular interventional surgery", *IEEE Sensors Journal*, vol.21, no.10, pp.12284-12294, DOI:10.1109/JSEN.2021.3066424, 2021.