Deep Reinforcement Learning for Vessel Centerline Tracing in Multi-modality 3D Volumes

Pengyue Zhang[1,2(✉)], Fusheng Wang[1], and Yefeng Zheng[2]

[1] Department of Computer Science, Stony Brook University, Stony Brook, USA
pengyue.zhang@stonybrook.edu
[2] Medical Imaging Technologies, Siemens Healthineers, Princeton, USA

Abstract. Accurate vessel centerline tracing greatly benefits vessel centerline geometry assessment and facilitates precise measurements of vessel diameters and lengths. However, cursive and longitudinal geometries of vessels make centerline tracing a challenging task in volumetric images. Treating the problem with traditional feature handcrafting is often ad-hoc and time-consuming, resulting in suboptimal solutions. In this work, we propose a unified end-to-end deep reinforcement learning approach for robust vessel centerline tracing in multi-modality 3D medical volumes. Instead of time-consuming exhaustive search in 3D space, we propose to learn an artificial agent to interact with surrounding environment and collect rewards from the interaction. A deep neural network is integrated to the system to predict stepwise action value for every possible actions. With this mechanism, the agent is able to probe through an optimal navigation path to trace the vessel centerline. Our proposed approach is evaluated on a dataset of over 2,000 3D volumes with diverse imaging modalities, including contrasted CT, non-contrasted CT, C-arm CT and MR images. The experimental results show that the proposed approach can handle large variations from vessel shape to imaging characteristics, with a tracing error as low as 3.28 mm and detection time as fast as 1.71 s per volume.

1 Introduction

Detection of blood vessels in medical images can facilitate the diagnosis, treatment and monitoring of vascular diseases. An important step in vessel detection is to extract their centerline representation that can streamline vessel specific visualization and quantitative assessment. Precise vascular segmentation and centerline detection can serve as a reliable pre-processing step that enables precise determination of the vascular anatomy or pathology, which can guide

Electronic supplementary material The online version of this chapter (https://doi.org/10.1007/978-3-030-00937-3_86) contains supplementary material, which is available to authorized users.

A. F. Frangi et al. (Eds.): MICCAI 2018, LNCS 11073, pp. 755–763, 2018.
https://doi.org/10.1007/978-3-030-00937-3_86

pre-surgery planning in vascular disease treatment. However, automatic vessel centerline tracing still faces several major challenges: (1) vascular structures constitute only a small portion of the medical volume; (2) vascular boundaries tend to be obscure, with presence of nearby touching anatomical structures; (3) vessel usually has an inconsistent tubular shape with changing cross-section area, which poses difficulty in segmentation; (4) it is often hard to trace a vessel due to its cursive lengthy structure.

Majority of existing centerline tracing techniques compute centerline paths by searching for a shortest path with various handcrafted vesselness or medialness cost metrics such as Hessian based vesselness [1], flux based medialness [2] or other tubularity measures along the paths. However, these methods are sensitive to the underlying cost metric. They can easily make shortcuts through nearby structures if the cost is high along the true path, which is likely to happen due to vascular lesions or imaging artifacts. Deep learning based approaches are proved to be able to provide better understanding from data and demonstrate superior performance compared to traditional pattern recognition methods with hand-crafted features. However, directly applying fully-supervised CNN with an exhaustive searching strategy is suboptimal and can result in inaccurate detection and huge computation time, since many local patches are not informative and can bring additional noise.

In this paper, we address the vessel centerline tracing problem with an end-to-end trainable deep reinforcement learning (DRL) network. An artificial agent is learned to interact with surrounding environment and collect rewards from the interaction. We can not only generate the vesselness map by training a classifier, but also learn to trace the centerline by training the artificial agent. The training samples are collected in such an intelligent way that the agent learns from its own mistakes when it explores the environment. Since the whole system is trained end-to-end, shortest path computation, which is used in all previous centerline tracing methods, is not required at all. Our artificial agent also learns when to stop. If the target end point of the centerline (e.g., iliac bifurcation for aorta tracing starting from the aortic valve) is inside the volume, our agent will stop there. If the target end point is outside of the volume, our agent follows the vessel centerline and stops at the position where the vessel goes out of the volume. Quantitative results demonstrate the superiority of our model on tracing the aorta on multimodal (including contrasted/non-contrasted CT, C-arm CT, and MRI) 3D volumes. The method is general and can be naturally applied to trace other vessels.

2 Background

Emerging from behavior psychology, reinforcement learning (RL) approaches aim to mimic humans and other animals to make timely decisions based on previous experience. In reinforcement learning setting, an artificial agent is learned to take actions in an environment to maximize a cumulative reward. Reinforcement learning problems consist of two sub-problems: the policy evaluation problem

which computes state-value or action-value function based on a given policy; and the control problem which searches for the optimal policy. These two sub-problems rely on the behavior of agent and environment, and can be solved alternatively.

Previously, reinforcement learning based approaches have achieved success in a variety of problems [3,4], but its applicability is limited to domains with fully observed and low dimensional spaces and its efficacy is bottlenecked by challenges in hand-crafted feature design in shallow models. Deep neural network can be integrated into reinforcement learning paradigm as a nonlinear approximator of value function or policy function. For example, a stabilized Q-network training framework was designed for AI game playing and demonstrated superior performance compared to previous shallow reinforcement learning approaches [5]. Following this work, several deep reinforcement learning based methods were proposed and made further improvements on game score and computing speed in game playing scenario [6,7]. Recently in [8,9], deep reinforcement learning framework was creatively leveraged to tackle important medical imaging tasks, such as 3D anatomical landmark detection and 3D medical image registration. In these methods, the medical imaging problems are reformulated as strategy learning process in a completely different way, in which artificial agents are trained to make sequential decisions and yield landmark detection or image alignment intelligently.

3 Method

In this section we propose a deep reinforcement learning based method for vessel centerline tracing in 3D volumes. Given a 3D volumetric image $\mathbf{I}$ and the list of ground truth vessel centerline points $\mathbf{G} = [\boldsymbol{g}_0, \boldsymbol{g}_1, \ldots, \boldsymbol{g}_n]$, we aim to learn a navigation model for an agent to trace the centerline through an optimal trajectory $\mathbf{P} = [\boldsymbol{p}_0, \boldsymbol{p}_1, \ldots, \boldsymbol{p}_m]$. We propose to solve the problem as a sequential decision making problem and model it as a reward-based Markov Decision Process (MDP). An agent is designed to interact with an environment over time. At each time step t, the agent receives state s from state space $\mathcal{S}$ and selects action a from action space $\mathcal{A}$ according to policy π. For vessel centerline tracing, we allow an agent to move to its adjacent voxels, resulting in an action space $\mathcal{A}$ with six actions $\{$*left, right, top, bottom, front, back*$\}$. A scalar reward $r_t = r_{s,a}^{s'}$ is used to measure the effect of the transition from state s to state s' through action a. To define the reward for centerline tracing, we first calculate minimum distance from the current point p_t to a point on the centerline and denote the corresponding point as g_d. Then, we define a point-to-curve distance-like measure:

$$D(\boldsymbol{p}_t, \mathbf{G}) = ||\lambda(\boldsymbol{p}_t - \boldsymbol{g}_{d+k}) + (1-\lambda)(\boldsymbol{g}_{d+k+1} - \boldsymbol{g}_{d+k-1})||. \tag{1}$$

This measure is composed of two components balanced by a scalar parameter λ, where the first component is pulling the agent position towards the ground truth centerline and the second one is a momentum enforcing the agent towards the

direction of the curve. k represents the forward index offset along the uniformly sampled curve (by default, $k = 1$). We also consider the reward calculation under two cases: when the current agent position is far from the curve, we want the agent to approach the curve as quickly as possible; when it is near the curve we also want it to move along the curve. Thus the step-wise reward is defined as

$$r_t = \begin{cases} D(\boldsymbol{p}_t, \mathbf{G}) - D(\boldsymbol{p}_{t+1}, \mathbf{G}), & \text{if } ||\boldsymbol{p}_t - \boldsymbol{g}_d|| <= l \\ ||\boldsymbol{p}_t - \boldsymbol{g}_d|| - ||\boldsymbol{p}_{t+1} - \boldsymbol{g}_d||, & \text{otherwise} \end{cases} \tag{2}$$

where l is an empirically chosen threshold for the point-to-curve distance. Note that when $l \to \infty$ and $\lambda = 1$ we have simplified forward distance based reward as $r_t = ||\boldsymbol{p}_t - \boldsymbol{g}_{d+k}|| - ||\boldsymbol{p}_{t+1} - \boldsymbol{g}_{d+k}||$.

We use the long-term expected return $R_t = \sum_{\tau=t}^{\infty} \gamma^{\tau-t} r_\tau$ as discounted accumulated reward with discount factor $\gamma < 1$. The action-value function $Q^\pi(s, a) = \mathbb{E}[R_t | s, a, \pi]$ represents the expected future discounted reward selecting a in state s and then following policy π. An optimal action-value function is defined as $Q^\star(s, a) = \max_\pi Q^\pi(s, a)$, which represents the reward collected by the agent which starts from state-action pair (s, a) and acts optimally thereafter. The corresponding optimal policy is $\pi^\star(s) = \arg\max_{a \in \mathcal{A}} Q^\star(s, a)$. By the Bellman equation [10], the optimal action-value function satisfies a recursive formulation:

$$Q^\star(s, a) = \mathbb{E}_{s'}[r + \gamma \max_{a'} Q^\star(s', a')]. \tag{3}$$

We parameterize and approximate the optimal action-value function by a deep neural network $Q(s, a; \theta) = Q^\star(s, a)$, where θ represents trainable parameters in the neural network. The optimal action-value target can be approximated as $y = r + \gamma \max_{a'} Q(s', a', \theta_{i'})$, where $\theta_{i'}$ is the network weights from some previous iteration $i' < i$. To avoid the correlation between sequence of observations which may cause instability in training, the target is updated every few iterations. Following the experience replay mechanism, we can cache a replay set D of length M and draw samples from D for network training. Then we can define the loss function as

$$\begin{aligned} \mathcal{L}_i(\theta_i) &= \mathbb{E}_{s,a,r}\left[\mathbb{E}_{s'}[y|s, a] - Q(s, a; \theta_i)\right] \\ &= \mathbb{E}_{s,a,r,s'}\left[y - Q(s, a; \theta_i)\right]^2 + \mathbb{E}_{s,a,r}\left[\mathbb{V}_{s'}[y]\right]. \end{aligned} \tag{4}$$

With fixed parameters $\theta_{i'}$ from previous iteration, we can calculate the gradient with respect to θ_i and apply stochastic gradient descent afterward:

$$\nabla_{\theta_i} \mathcal{L}(\theta_i) = \mathbb{E}_{s,a,r,s'}\Big[\Big(r + \gamma \max_{a'} Q(s', a', \theta_{i'}) - Q(s, a; \theta_i)\Big) \nabla_{\theta_i} Q(s, a; \theta_i)\Big]. \tag{5}$$

Training Details. For training reinforcement learning models in medical imaging problems, it is important to find a good probing strategy to avoid early overfitting and make the model robust. We train the model in an episodic manner, in which we start from one sample volume and accumulate samples in experience replay set. Then, we calculate a maximum returning action value based on

the current neural network among all six possible actions. We apply a ϵ-greedy policy that takes the greedy action with probability ϵ and a random action with probability $1-\epsilon$. To encourage exploration in early training epochs, we set ϵ as 1 at first and let it decrease to 0 at constant rate over training iterations. We also select starting point in a similar probabilistic way: given the ground truth path $\mathbf{G} = [\boldsymbol{g}_0, \boldsymbol{g}_1, \ldots, \boldsymbol{g}_n]$, we set the initial point $\boldsymbol{p}'_0$ as $\boldsymbol{g}_0$ with probability η and some random point along $\mathbf{G}$ with probability $1-\eta$. Furthermore, we randomly select the starting point $\boldsymbol{p}_0$ in a local patch centered at $\boldsymbol{p}'_0$ with size $10 \times 10 \times 10$ voxels. The agent reaches a termination state in an episode when it reaches the last point in ground truth path $\mathbf{G}$ or the step number reaches the maximum episode length. Then, we starts a new episode in another volume.

Vascular Centerline Tracing. With an unseen test sample, we provide a starting point $\boldsymbol{p}_0 = \boldsymbol{g}_0$ at the vascular root to the system. We set the state as local volume observation $s_0 = \mathbf{I}_{\boldsymbol{p}_0}$ which is a 3D patch centered at $\boldsymbol{p}_0$, and feed it into the detection model. From the neural network we generate an action a_0 which moves the current point to $\boldsymbol{p}_1$. Then, the current state is updated as $s_1 = \mathbf{I}_{\boldsymbol{p}_1}$ and fed into the neural network to generate action again. We repeat this process until the path converges on oscillatory-like cycles. To further stabilize the tracing process, we also apply momentum on action-values from network output: $r_t \leftarrow \alpha r_{t-1} + (1-\alpha) r_t$, where α is the momentum factor. The centerline tracing process stops if the agent moves out of the volume or if a cycle is formed, i.e., moving to a position already visited previously. We remove the cycle from the traced centerline path during detection.

We define a curve-to-curve distance metric to measure the tracing error. In our problem setting, the ground truth $\mathbf{G}$ consists of a list of 3D points $\boldsymbol{g}_i$ and the centerline is approximately represented as the set of concatenating segments $\mathbf{C} = \{c_{i,i+1}\}$ of adjacent points $\boldsymbol{g}_i$ and $\boldsymbol{g}_{i+1}$. We first compute the distance from a detected point $\boldsymbol{p}_j \in \mathbf{P}$ to the ground truth $\mathbf{G}$ by finding the minimum distance from $\boldsymbol{p}_j$ to any segments $c_{i,i+1} \in \mathbf{C}$ or points $\boldsymbol{g}_i \in \mathbf{G}$. Then, the distance from $\mathbf{P}$ to $\mathbf{G}$ is computed as the average distance from any point $\boldsymbol{p}_j \in \mathbf{P}$ to $\mathbf{G}$. The distance from $\mathbf{G}$ to $\mathbf{P}$ can be computed similarly and the curve-to-curve distance error is defined as the average of these two distances.

4 Experiment

4.1 Dataset

We evaluate the proposed approach on the problem of tracing centerline of thoracic/abdominal aorta. We collected a dataset of 531 contrasted CT, 887 non-contrasted CT, 737 C-arm CT and 232 MR volumes from multiple sites over the world. These data represent different imaging modalities, scopes and qualities. All of the volumes are normalized to 2 mm isotropic resolution before experiments. We also map the intensity distribution of MR volumes to CT to make

sure they are equally bright. From the original 12-bit images, we clip and normalize the voxel intensities within [500, 2000]. We mix all volumes from different modalities and partition the dataset into training set and test set with 3:1 ratio on each modality. Ground truth annotations are provided by experts and reviewed by different people to ensure correctness.

4.2 Network Architecture and Implementation

We use a multi-layer neural network as a non-linear approximator for the action value function. The network consists of several convolutional, batch normalization, and fully connected layers. The first hidden layer is a convolutional layer with 32 filters of size $4 \times 4 \times 4$ and stride 2 followed by a batch normalization layer and a ReLU nonlinearity layer. The second hidden layer is a convolutional layer with 46 filters of size $3 \times 3 \times 3$ and stride 2. The following layers are two fully connected layers with 256 and 128 units, respectively. The last layer is also a fully connected layer with a probabilistic output for six possible actions.

The experiments was conducted on a server with one Nvidia Titan X GPU. We trained the model for 3000 epochs which takes about 64 h with an average running time of 77.13 s per epoch. The target network parameters were frozen and updated every 10,000 iterations. We used a set of 100,000 samples to store the history samples. The batch size was 8 and the learning rate was 0.0005 throughout the training process. The forward offset k was set as 1 and the detection momentum was set as $\alpha = 0.8$ based on our experiment. Other

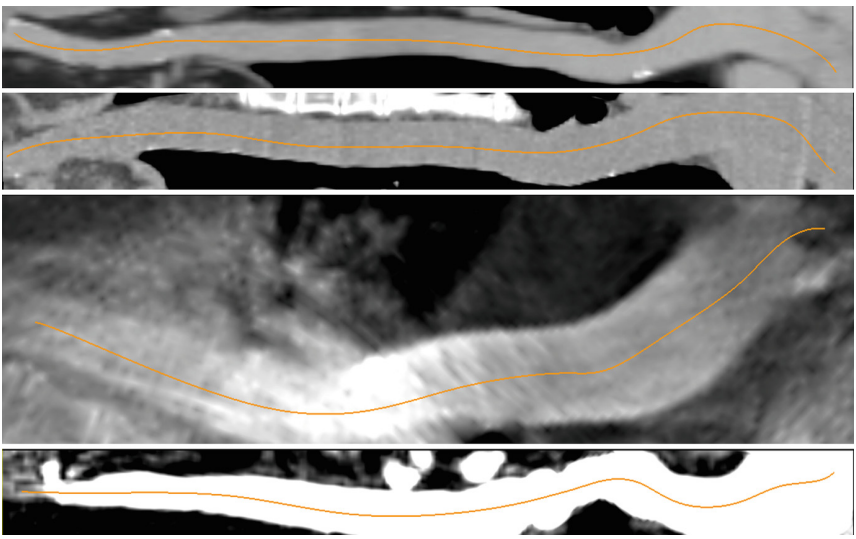

Fig. 1. Examples of traced aorta centerlines in the curved planar reformatting (CPR) view. From top to bottom: contrasted CT, non-contrasted CT, C-arm CT and MR. We recommend the readers to refer to the videos in supplementary material for better visualization effects.

parameters were set empirically as $\lambda = 0.5, \gamma = 0.9, \eta = 0.5$. Noticing that the exploration trend was gradually suppressed as the number of training iteration increased, we also gradually decreased the maximum length of each episode. The detected curve was represented as integer coordinates and then smoothed by B-spline interpolation. Over our experiment, we used volume patches with $50 \times 50 \times 50$ voxels as processing units.

4.3 Evaluation and Discussion

We evaluate the proposed deep reinforcement learning based 3D vessel centerline tracing approach on 3D medical volumes. The vessel centerline tracing results of our method is illustrated in Fig. 1. We observe that our deep reinforcement learning based model can trace the vessel centerline precisely. More importantly,

Table 1. Quantitative evaluation of different methods measured by the curve-to-curve distance in mm. A volume is considered as a failed case if the curve-to-curve distance is larger than 10.0 mm.

Modality	Method	Mean	Median	Std	80 percentile	Max	% failed
Contrasted CT	SL-CNN	8.62	4.67	10.94	5.35	49.38	11.11%
	DRL-1	5.71	5.43	2.42	5.90	25.67	1.39%
	DRL-2	4.77	4.79	0.44	5.15	6.04	**0%**
	DRL-3	4.04	**2.89**	5.54	3.26	38.30	2.78%
	DRL-4	**2.94**	2.93	**0.36**	**3.22**	**4.10**	**0%**
Non-contrasted CT	SL-CNN	4.84	4.59	2.31	4.91	33.78	1.35%
	DRL-1	4.98	4.97	0.52	5.34	7.32	**0%**
	DRL-2	4.75	4.75	0.43	5.11	6.18	**0%**
	DRL-3	3.04	3.01	**0.33**	3.27	**5.13**	**0%**
	DRL-4	**3.00**	**2.93**	0.65	**3.19**	11.31	0.45%
C-arm CT	SL-CNN	7.35	4.77	9.06	6.17	55.82	9.77%
	DRL-1	5.93	5.30	4.42	6.22	47.26	2.73%
	DRL-2	5.13	4.78	**2.85**	5.64	35.26	**1.17%**
	DRL-3	4.23	3.11	4.73	4.39	38.29	3.13%
	DRL-4	**3.72**	**3.09**	2.90	**4.18**	**33.08**	1.56%
MR	SL-CNN	14.85	6.17	11.86	27.44	40.49	43.10%
	DRL-1	6.68	5.85	**2.67**	7.37	**20.40**	8.48%
	DRL-2	6.56	5.20	5.47	6.00	30.73	**3.39%**
	DRL-3	**5.09**	3.31	5.89	**3.81**	29.38	6.78%
	DRL-4	5.51	**3.30**	6.89	4.09	38.88	8.48%
Overall	SL-CNN	7.07	4.64	8.18	5.31	55.82	9.53%
	DRL-1	5.63	5.09	3.73	5.82	47.26	2.43%
	DRL-2	5.06	4.78	**2.53**	5.30	**35.26**	**0.94%**
	DRL-3	4.02	3.17	4.40	3.69	38.30	2.43%
	DRL-4	**3.23**	**2.81**	2.86	**3.35**	38.88	1.86%

our method has nice generalization property and performs consistently over different imaging modalities. We compare the proposed method with a supervised 3D convolutional neural network (SL-CNN) based approach which shares the same network architecture with the proposed DRL method in Table 1. However, SL-CNN is trained with uniformly sampled patches from the training volume to predict moving actions as output labels. We apply the same detection process and hyper parameters for fair comparison. So, the only difference between the SL-CNN approach and DRL approach is how the action network is trained. We consider four variants of the proposed DRL methods with slightly different settings: For DRL-1, DRL-2 and DRL-3 we use the reward function with momentum. In DRL-1 we remove the vessel radius limit by setting $l = \infty$, while in DRL-2 and DRL-3 we set $l = 4$ and $l = 2$, respectively. In DRL-4, we remove the momentum term and simply use the forward distance based reward. The curve-to-curve distance is used to evaluate tracing accuracy of an algorithm. We observe that all the DRL based method can outperform SL-CNN by a considerable margin and they perform consistently over different imaging modalities. We can also observe that the proposed DRL-4 method with forward distance based reward function can achieve best tracing error while DRL-2 with momentum reward has fewest failed cases. The results also demonstrate that setting vessel diameter threshold parameter l can potentially improve the tracing performance. By using smaller vessel diameter threshold l as in DRL-3, the agent can trace the curve in a finer way but it is also more prone to early stop. DRL-2 will be used in practice since reducing failure rate is more desired for our vessel centerline tracing task. With the proposed DRL based method, we can provide fast vessel centerline extraction, with an average detection time of 1.71 s per volume.

5 Conclusion

In this paper, we propose a deep reinforcement learning approach for vessel centerline tracing in 3D medical volumes. By reformulating the problem as a behavior learning problem, we establish an interactive reinforcement learning model to train an artificial agent. The agent communicates with the surrounding environment and receives feedback from the environment to guide action selection in next steps. Using a deep neural network as a non-linear approximator for the action-value function, we can train the model in an end-to-end manner without any requirements for feature engineering. The proposed method is evaluated on over 2,000 3D medical volumes with four different modalities and demonstrates satisfying performance on all of the modalities.

Acknowledgement. This research is supported in part by grants from National Science Foundation ACI 1443054 and IIS 1350885.

References

1. Frangi, A.F., Niessen, W.J., Vincken, K.L., Viergever, M.A.: Multiscale vessel enhancement filtering. In: Wells, W.M., Colchester, A., Delp, S. (eds.) MICCAI 1998. LNCS, vol. 1496, pp. 130–137. Springer, Heidelberg (1998). https://doi.org/10.1007/BFb0056195
2. Siddiqi, K., Bouix, S., Tannenbaum, A., Zucker, S.: Hamilton-Jacobi skeletons. Int. J. Comput. Vis. **48**(3), 215–231 (2002)
3. Riedmiller, M., Gabel, T., Hafner, R., Lange, S.: Reinforcement learning for robot soccer. Auton. Robots **27**(1), 55–73 (2009)
4. Diuk, C., Cohen, A., Littman, M.L.: An object-oriented representation for efficient reinforcement learning. In: Proceedings of the 25th International Conference on Machine Learning, pp. 240–247, ACM (2008)
5. Mnih, V., et al.: Human-level control through deep reinforcement learning. Nature **518**(7540), 529 (2015)
6. Mnih, V., et al.: Asynchronous methods for deep reinforcement learning. In: International Conference on Machine Learning, pp. 1928–1937 (2016)
7. Silver, D., et al.: Mastering the game of Go with deep neural networks and tree search. Nature **529**(7587), 484–489 (2016)
8. Ghesu, F.C., Georgescu, B., Mansi, T., Neumann, D., Hornegger, J., Comaniciu, D.: An artificial agent for anatomical landmark detection in medical images. In: Ourselin, S., Joskowicz, L., Sabuncu, M.R., Unal, G., Wells, W. (eds.) MICCAI 2016. LNCS, vol. 9902, pp. 229–237. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46726-9_27
9. Liao, R., et al.: An artificial agent for robust image registration. In: AAAI, pp. 4168–4175 (2017)
10. Bellman, R.: Dynamic Programming. Courier Corporation, North Chelmsford (2013)