Determining scanning trajectories for robot-CT using reinforcement learning

Seokwon Oh, Seungjun Yoo, Junho Lee, Seongbon Park, and Ho Kyung Kim*
Computational X-ray Imaging Laboratory, School of Mechanical Engineering, Pusan Nat'l Univ., Busandaehakro 63beon-gil, Busan 46241
**Corresponding author: hokyung@pusan.ac.kr*

***Keywords: Trajectory optimization, Reinforcement learning, Vision transformer**

1. Introduction

Computed tomography (CT) is a powerful non-invasive technique for visualizing internal structures and has been widely utilized in the medical field. Recently, with the growing demand for industrial CT, various efforts have been made to adapt CT technology for industrial applications. However, conventional circular scanning methods often fail to guarantee sufficient image quality when faced with challenges such as artifacts from large or metallic objects. To overcome these limitations, a novel approach has been proposed that employs an x-ray source and a detector mounted on robotic arms to perform three-dimensional scanning, along with several methods to optimize the scanning trajectory for high-quality reconstruction of a volume of interest (VOI) [1-2].

Traditional studies have primarily relied on pre-existing CAD data or historical datasets to simulate scanning from all angles in three-dimensional space. These simulations involve calculating metrics such as detectability and transmittance from various angles to determine an optimal scanning trajectory for effective VOI reconstruction. However, these methods are limited in scenarios where pre-existing data is unavailable, the acquired projection image differs from simulated data, or when it is necessary to determine an optimal trajectory in real time with current projection image.

In this study, we propose an approach that integrates the vision transformer (ViT) [3] with the dueling double deep Q-network architecture (DQN) [4-6]. This framework effectively fuses VOI-masked image information with projection images and employs reinforcement learning to learn the optimal scanning path autonomously.

2. Methods and Materials

2.1. CT system with dual robotic arms

In this study, the CT system is configured with two robotic arms, each equipped with an x-ray source and a detector, and arranged to maintain a fixed distance from the target object while the x-ray source acquires projection images along a spherical coordinate system (φ_t: azimuthal angle, θ_t: polar angle). Here, t is defined as the projection angle index, and φ_t is defined as $\varphi_t = 2(t-1)°$ for $t \in \{1,2,\dots,180\}$, while θ_t denotes the θ value at t. The objective is to train a network to select θ_{t+1}, which enables effective reconstruction of the VOI at φ_{t+1} using the projection image from (φ_t, θ_t). The θ values are 45°, 60°, and 90°, and the action ($a \in \mathcal{A}$) is to choose one of these three θ values. The current state s_t is composed of four elements $(P_t, M_t, \varphi_t, \theta_t)$. The P_t is the projection image acquired at the (φ_t, θ_t) angles, and M_t is a masking image produced by multiplying P_t with the projection result of only the VOI region at the same angle.

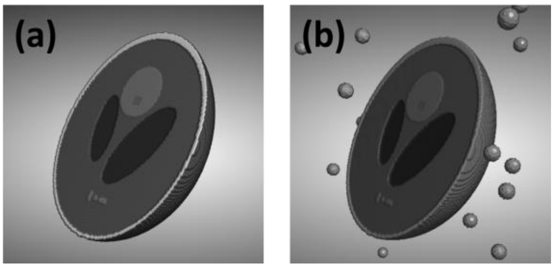

Fig. 1. Two types of phantoms used in this study. The left (a) is the original phantom, and the right (b) is the randomly beaded phantom.

For network training, a modified Shepp-Logan phantom with internal properties altered (e.g., aluminum, water, synthetic resin, etc.) was used, and the x-ray energy was set to 70 kV. To make an environment that hinders VOI reconstruction, the projection image of the two types of phantoms was used for training: the randomly beaded phantom (RB phantom), in which steel beads were randomly placed around the phantom as shown in Fig. 1(a), and the original phantom without beads as shown in Fig. 1(b). Our goal is to reconstruct the VOI of the RB phantom with high quality. For the RB phantom, the projection image corresponds to $\theta \in \{45°, 60°, 90°\}$, while the original phantom consists only of the projection image at $\theta = 90°$.

2.2. Network structure and reinforcement learning

In the network model, φ and θ are each passed through a linear layer to be transformed into high-dimensional embeddings, which are then summed. This conditional embedding is added to P_{token} and M_{token} with positional embedding applied, thereby reinforcing information about the angles at which the input images P and M were acquired. The M_{token} is used as the Query, while P_{token} is used as the Key and Value in performing cross-attention. This is intended so that the M_{token}, which contains information about the VOI, can reference

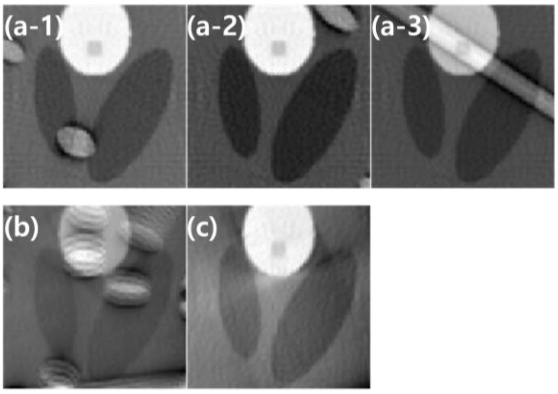

Fig. 2. (a) Reconstructed images reconstructed with FDK algorithm with original projection images except at the angle at φ_{t+1}. At that specific angle, the projection image was replaced by the RB phantom's projection at different θ values—(a-1) $\theta = 45°$, (a-2) $\theta = 60°$, and (a-3) $\theta = 90°$. (b) Reconstructed image using FDK with the projection at each φ chosen as the one yielding the highest PSNR among the θ values. (c) Improved reconstruction image achieved by excluding the problematic angle φ_{t+1} when all θ causes artifacts.

P_{token} as the key and value to avoid angles that would degrade the quality of the VOI.

After converting the output of the Transformer into a one-dimensional vector, it is split into forms that yield the state value (V) and the advantage (A). Then, using the V and A, the Q-value is calculated for each action as in Eq. (1).

$$Q_w(s_t, a_t) = V_w(s_t) + \left(A_w(s_t, a_t) - \frac{1}{|\mathcal{A}|}\sum_{a' \in \mathcal{A}} A_w(s_t, a')\right) \tag{1}$$

Based on double DQN, a target value (y_t) is set as defined in Eq. (2), and the network parameters (w) are trained. At the state (s_t), the optimal action is estimated using the online network Q_w, and the Q-value for that action is retrieved from the target network Q_{w^-}. This Q-value is then multiplied with the discount factor γ and summed with a reward (r_{t+1}) to calculate the target value. The network is trained to minimize the difference between this target value and $Q_w(s_t, a_t)$.

$$y_t = r_{t+1} + \gamma Q_{w^-}\left(s_{t+1}, \underset{a'}{\operatorname{argmax}}\, Q_w(S_{t+1}, a')\right) \tag{2}$$

The reward was computed as follows. First, based on the Q-value, a next state $(P_{t+1}, M_{t+1}, \varphi_{t+1}, \theta_{t+1})$ is determined. Then, the projection image of the original phantom at that angle is replaced with the projection image (P_{t+1}) of the RB phantom at the same angle to reconstruct the VOI$'$. Next, the PSNR for each slice is calculated between the VOI$'$ and the VOI reconstructed using the original phantom's projection images; the average of PSNR values is used as the reward. With this computation of the reward, the impact of the projection image obtained at $(\varphi_{t+1}, \theta_{t+1})$ produced by the network on the VOI can be evaluated, as shown in Fig. 2(a). Therefore, one might consider selecting the θ_t that results in a high PSNR at each φ_t to achieve a good reconstruction of the VOI. However, if all θ projection images at a given φ_t induce artifacts in the VOI as in Fig. 2(a), the quality of the reconstructed image may deteriorate as Fig. 2(b). To resolve this issue, we employed a method that skips the corresponding φ_{t+1} if all θ values at that φ_{t+1} cause artifacts; Fig. 2(c) shows the result of this method. To mimic this in the network, in addition to outputting the state value function and advantage, the network is designed to output the φ angle to be skipped, and this is indirectly learned through the reward.

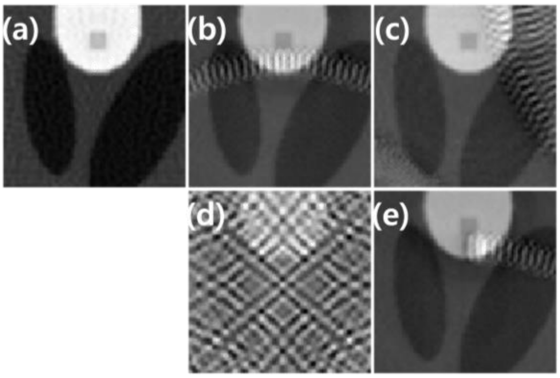

Fig. 3. (a) Reconstructed image from projection images of original phantom ($\theta = 90°$). (b-d) Reconstructed images from projection images of RB phantom ($\theta = 45°$, $60°$, $90°$). (e) Reconstructed image from the network-based optimized trajectory.

3. Preliminary Results

Fig. 3 demonstrates the reconstruction results of the original phantom (Fig. 3(a)) and the RB phantom (Fig. 3(b-e)). Fig. 3(d) is the result of the conventional trajectory ($\theta = 90°$), which shows the worst result. Fig. 3(b) and (c) are the reconstruction results of $\theta = 45°$ and $60°$. Fig. 3(e) is an image reconstructed from a network-based optimized trajectory with 176 projection images. This VOI showed the best result in respect of SSIM (= 0.89).

4. Conclusion

This study presents a method that combines the ViT with the dueling double DQN to optimize scanning trajectories. Preliminary results show that reinforcement learning may offer a practical solution for addressing challenges in CT trajectory optimization. The network will be validated with real experimental data obtained from a single-arm robot system that produces equivalent results to those originally implemented on a dual-robotic-arm CT.

ACKNOWLEDGEMENTS

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. RS-2024-00340520) and supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (No. RS-2024-00408137).

REFERENCES

[1] G. Herl, J. Hiller, and A. Maier, "Scanning trajectory optimisation using a quantitative Tuybased local quality estimation for robot-based X-ray computed tomography," Nondestructive Testing and Evaluation, Vol. 35, No. 3, pp. 287–303, 2020.

[2] A. Fischer, T. Lasser, M. Schrapp, J. Stephan, and P. B. Noël, "Object Specific Trajectory Optimization for Industrial X-ray Computed Tomography," Scientific Reports, Vol. 6, No. 1, pp. 19135, 2016.

[3] A. Dosovitskiy, L. Beyer, A. Kolesnikov, et al., "An Image is Worth 16×16 Words: Transformers for Image Recognition at Scale," arXiv preprint arXiv:2010.11929, 2020.

[4] V. Mnih, K. Kavukcuoglu, D. Silver, et al., "Playing Atari with Deep Reinforcement Learning," arXiv preprint arXiv:1312.5602, 2013.

[5] H. van Hasselt, A. Guez, and D. Silver, "Deep Reinforcement Learning with Double Q-learning," arXiv preprint arXiv:1509.06461, 2015.

[6] Z. Wang, T. Schaul, M. Hessel, et al, "Dueling Network Architectures for Deep Reinforcement Learning," arXiv preprint arXiv:1511.06577, 2016.