Gaze-Based Dual Resolution Deep Imitation Learning for High-Precision Dexterous Robot Manipulation

Heecheol Kim, Yoshiyuki Ohmura, and Yasuo Kuniyoshi

***Abstract*—A high-precision manipulation task, such as needle threading, is challenging. Physiological studies have proposed connecting low-resolution peripheral vision and fast movement to transport the hand into the vicinity of an object, and using high-resolution foveated vision to achieve the accurate homing of the hand to the object. The results of this study demonstrate that a deep imitation learning based method, inspired by the gaze-based dual resolution visuomotor control system in humans, can solve the needle threading task. First, we recorded the gaze movements of a human operator who was teleoperating a robot. Then, we used only a high-resolution image around the gaze to precisely control the thread position when it was close to the target. We used a low-resolution peripheral image to reach the vicinity of the target. The experimental results obtained in this study demonstrate that the proposed method enables precise manipulation tasks using a general-purpose robot manipulator and improves computational efficiency.**

***Index Terms*—Imitation learning, deep learning in grasping and manipulation, bioinspired robot learning, telerobotics and teleoperation, failure detection and recovery.**

I. Introduction

DEEP imitation learning ([1], [2]) is used to train deep neural networks on demonstration data, and has good potential for application to robots used in daily life because it does not require a hard-coded robot control rule. This study demonstrates that deep imitation learning can be applied to high-precision visuomotor robot manipulation tasks, such as needle threading. This task is difficult both for humans and robots because (1) the clearance is small, (2) the thread is deformable, and (3) the posture of the needle varies in each picking trial. Therefore, this task requires a complex control policy with high precision. If it can be demonstrated that the general-purpose robot manipulator can learn a complex policy, the scope of future robots used in daily life will expand.

Humans use two control systems with different visual resolutions for manipulation tasks [3]. The anatomical study of the human eye indicates that the retina is divided into the fovea and peripheral vision. The fovea is located at the center of the retina with a cone photoreceptor-dominated field, which is denser and relatively thicker than other parts of the retina [4], and thus provides high-resolution visual information. When a human moves its hand to a target, the eye gaze is naturally foveated at the target [5]–[8], whereas the peripheral field contains a relatively sparse placement of photoreceptors, mostly dominated by rods [4]. Central vision and peripheral vision are functionally separated when the hand reaches out to the target under visual guidance [3]. The human first moves its hand into the vicinity of the target with fast feedback loop control using peripheral visual information, and accomplishes the accurate homing of the target through a slow feedback loop with central vision at the end of the manipulation trajectory [3], [9]. However, most existing deep learning based visuomotor control methods for robot manipulation learn an action policy with the entire visual input [1], [2], [10], [11]. A recent study [12] used human gaze during demonstrations to learn the action policy. However, the visuomotor control system was not separated into foveated vision and peripheral vision.

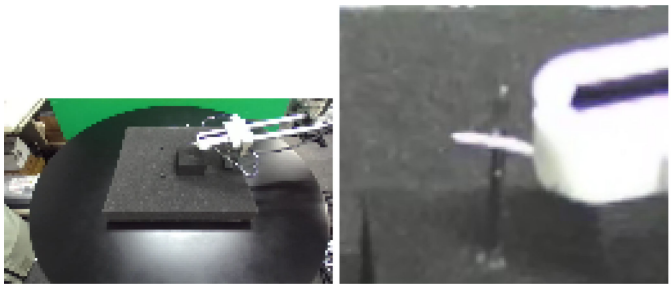
This study proposes a highly precise object manipulation method inspired by the separated visuomotor control system of humans. The foveated vision (10°) around the gaze position measured by the eye-tracker retains high spatial resolution (142×120 from the 1280×720 entire image), and the entire image is resized as a low-resolution image (128×72) to form the peripheral vision (Fig. 1). The deep neural network model infers fast-action only with a peripheral image to reach the target, and slow-action only with a foveated image to precisely grasp the thread or insert it into the eye of the needle. This separated visuomotor control architecture has two benefits compared with previous deep imitation learning methods. First, using the foveated vision with an explicit gaze mechanism improves generalization because (1) this mechanism extracts important visual features to focus on from the entire scene, and (2) fewer neural network parameters are required. Second, higher computational efficiency can be achieved because only the foveated vision retains high-resolution.

Manuscript received October 15, 2020; accepted January 25, 2021. Date of publication February 16, 2021; date of current version March 2, 2021. This work was supported in part by the Grant-in-Aid for Scientific Research (A) JP18H04108 and in part by the Project commissioned by the New Energy and Industrial Technology Development Organization (NEDO). *(Corresponding author: Heecheol Kim.)*

The authors are with the Laboratory for Intelligent Systems and Informatics, Graduate School of Information Science and Technology, The University of Tokyo, Tokyo 113-0023, Japan (e-mail: h-kim@isi.imi.i.u-tokyo.ac.jp; ohmura@isi.imi.i.u-tokyo.ac.jp; kuniyosh@isi.imi.i.u-tokyo.ac.jp).

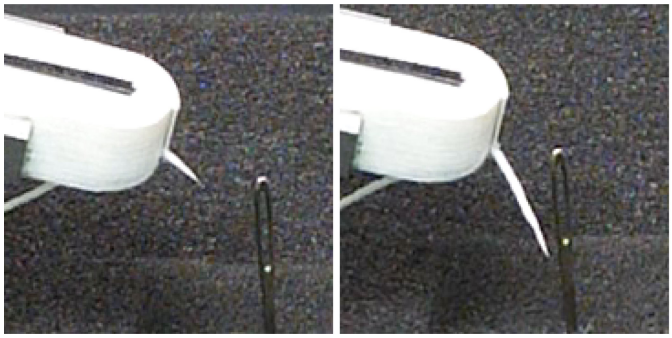
This letter has supplementary downloadable material available at https://doi.org/10.1109/LRA.2021.3059619, provided by the authors.

Digital Object Identifier 10.1109/LRA.2021.3059619



(a) Low-resolution peripheral vision (b) High-resolution foveated vision

Fig. 1. The proposed method can efficiently calculate a precise policy with both global features from peripheral vision (1a) and detailed visual information for the needle and thread from foveated vision (1b).



(a) Short grasp (b) Long grasp

Fig. 2. Difference between thread grasps. The proposed method can adjust its policy with respect to the posture of the grasped thread.

II. Related Work

One representative task of small clearance manipulation is the peg-in-hole task. Because this task has small clearance, it is important to establish a control model of the peg under friction with the hole [13] or achieve the accurate alignment of the peg and hole [14]–[17]. Force/torque feedback is mainly used to estimate the state of the peg/hole or the friction between the peg and the hole [13], [18], [19]. [17] used visual feedback to align the peg and hole. By contrast, needle threading cannot use force feedback because the robot cannot sense friction between the needle and the flexible thread. Therefore, unlike the peg-in-hole which handles rigid objects, the robot must adjust its policy with thread deformation using visual feedback. Because it is difficult to establish a model with high-precision deformation prediction, the model-based approach cannot be applied to the needle threading task.

Related studies, such as [20], [21] and [22], considered needle threading tasks. [20] achieved a rope-into-ring task. An imaginary guide was set inside the ring to insert the rope tip into the ring. For the needle threading task considered in this study, it is difficult to calculate the imaginary guide tube because of the small clearance and insufficient visual resolution, even with 1280×720 images. Furthermore, in our setup, the robot must appropriately control the thread, even when the thread is deformed by collision with the needle. [21] achieved high-precision assembly behaviors with laser-based sensing and validated with needle threading and USB insertion tasks. A dual-arm system where one sensing arm mounts the laser scanner was used to compensate for errors. However, their approach requires a high-resolution depth sensing device mounted on a robot arm, which may not be suitable for a general-purpose robot system, and assumes that the thread is rigid. [22] achieved needle threading using high-speed visual feedback. Their idea was that a thread rotating with high-speed can be approximated as a rigid object, and thus the control models can be simplified. However, this study used a two-degree-of-freedom single purpose mechanism and did not automate the thread mounting. To the best of our knowledge, a series of pick-and-threading tasks has not been learned by a general-purpose robot manipulator to date.

Active vision is a research field about actively control the camera coordinate with respect to given sensory input to focus visual attention to a relevant target [23]–[25]. [26], [27] achieved active stereo vision on a camera head system. [28]–[30] designed camera systems in which high-resolution central vision and low-resolution peripheral vision in hardware, and [29], [30] further implemented the proposed camera lens on an active stereo vision system. [31], [32] extended attention to sensory-motor control. [31] proposed a reinforcement learning system that can learn to focus on necessary sensory input. [32] used foveated images for imitation learning on a two-dimensional video game. However, our research objective is to demonstrate that visual attention acquired from the human gaze can improve performance for real-world robot manipulation.

III. Method

A. High-Resolution Image Processing With Human Gaze

In our setup, a human operator teleoperates a UR5 robot (Universal Robots) based on visual input from a head-mounted display (HMD), which reflects images from the ZED Mini stereo camera [33] on the robot, while the eye tracker mounted on the HMD measures the operator's eye gaze.

The high definition (HD) stereo image (1280×720) from the camera is processed into a foveated image and peripheral image. The peripheral image is the entire stereo image reduced to 128×72. The foveated image is a stereo image cropped to 142×120 around both the left and right gaze from the raw HD image, and corresponds to the central retina (10°) of the human eye [3] (see Appendix A for details). The derived foveated and peripheral images are used as input for the visuomotor policy calculation introduced in III-D. Each image and subsequent robot joint angles are collected at 10 Hz. Therefore, each step is defined as 0.1 seconds throughout this letter.

B. Action Separation by Speed of Action

The target-reaching movement of humans is divided into the fast reaching movement calculated from peripheral vision and accurate homing with a slow feedback loop calculated from central vision [3], [9]. In our method, on the basis of this result obtained from physiological studies, the separated foveated and

Algorithm 1: Proposed Algorithm.

Parameter: Fast-action π_{fast}, slow-action (standard) π_{slow}, slow-action (recovery) π_{rec}, gripper open/close π_{gp}, recovery classifier θ, gaze predictor ρ, recovery step predictor ψ, action speed classifier ϕ

$step \leftarrow 0$
$count \leftarrow 0$
$succeed \leftarrow$ false
while $step < 500$ **And** $\neg\, succeed$ **do**
  $o_l, o_r \leftarrow 1280 \times 720$ left/right camera image
  $p_l, p_r \leftarrow resize(o_l, (128, 72)), resize(o_r, (128, 72))$ {resize to left/right peripheral vision p_l, p_r}
  $p \leftarrow concat(p_l, p_r)$ {concatenate peripheral vision}
  $g_l, g_r \leftarrow \rho(p)$ {predict left/right gaze position g_l, g_r}
  $c_l, c_r \leftarrow crop(o_l, g_l), crop(o_r, g_r)$ {crop left/right foveated vision}
  $c \leftarrow concat(c_l, c_r)$ {concatenate foveated vision}
  $gp \leftarrow \pi_{gp}(c)$ {predict gripper command gp}
  if $\theta(c)$ is true **And** $count = 0$ **then**
    $count \leftarrow \psi(c)$
  end if
  if $\phi(c)$ is fast-action **then**
    $action \leftarrow \pi_{fast}(p)$ {predict action from peripheral vision}
  else
    if $count > 0$ **then**
      $action \leftarrow \pi_{rec}(c)$ {predict recovery action from foveated vision}
      $count \leftarrow count - 1$
    else
      $action \leftarrow \pi_{slow}(c)$ {predict action from foveated vision}
    end if
  end if
  Execute $(action, gp)$ on robot
  $step \leftarrow step + 1$
  Manually decide $succeed$
end while

peripheral images are used to infer slow-action and fast-action, respectively. As rotation is not significant in our needle threading and bolt picking tasks, the action speed is defined by the Euclidean norm of the positional difference (1), whereas the action is defined as the difference in the end-effector position and orientation between the next step and current step during the teleoperation:

$$s_t = \sqrt{(x_{t+1} - x_t)^2 + (y_{t+1} - y_t)^2 + (z_{t+1} - z_t)^2}, \quad (1)$$

where s_t represents the speed of action at timestep t. The action is separated into fast-action and slow-action according to the speed of the action using a threshold. To determine the threshold that separates fast-action and slow-action, we assume that fast-action and slow-action are sampled from mutually independent probability distributions in terms of the action speed. A Gaussian mixture model (GMM) with two Gaussian distributions corresponding to the slow-action and fast-action, respectively, is fitted from the $[x, y, z]$ speed of actions (Fig. 3). Then, the intersection point of the two Gaussian distributions is determined as the threshold.

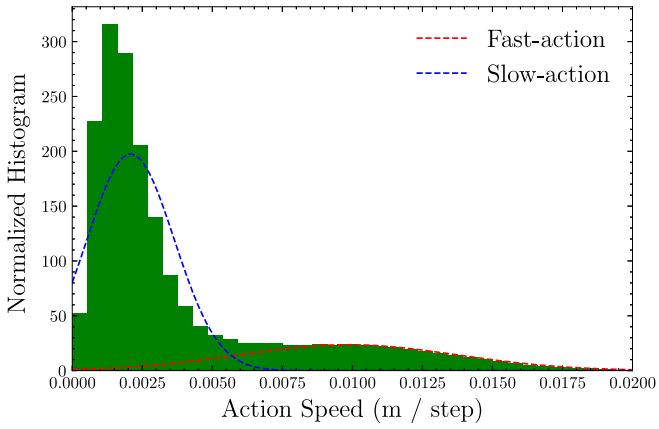


Fig. 3. Histogram of action speed and fitted GMM of needle threading. The intersection point of the two Gaussian distributions is defined as the threshold between slow-action and fast-action.

After learning, the action has to be automatically selected from either slow-action or fast-action. To achieve this, a binary classifier, that is, the *action speed classifier*, is trained using the action speed labels. The foveated vision is used as the input to the *action speed classifier* because we observed that the foveated vision had sufficient information for classification.

C. Recovery Action

Owing to the small clearance of the needle threading task, even human demonstrators fail frequently. In the training data, the failure rate on each trial was 36.4%. In these failures, humans recover from the failure and try again until the threading is finally successful (Fig. 5). Hence, we assume that the robot should be able to recover from failure and achieve better performance in subsequent attempts.

In this study, we adopt neural networks that recognize the failure state and switch to the recovery action (Fig. 4). The failure states and recovery actions are annotated based on demonstration data, respectively. The failure states are defined as threads passing by the needle without being inserted into the eye of the needle, whereas the recovery actions are defined as retreating actions until the next threading. Failure actions are removed from the training data.

The annotated recovery actions are separated from other threading actions in the training data (standard action) and are trained in a separate neural network that only infers the recovery action. Two additional neural networks are used to recognize the failure state. From the current foveated image, the *recovery classifier* predicts whether the current state is a failure, and the *recovery step predictor* predicts how many steps are required to complete the recovery. If the current state is a failure, the robot

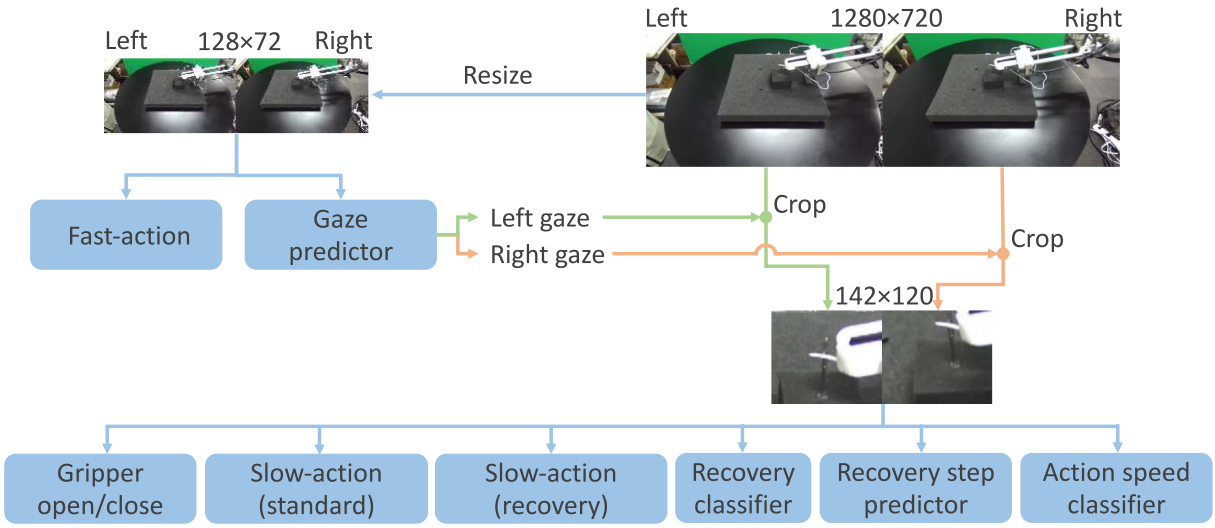
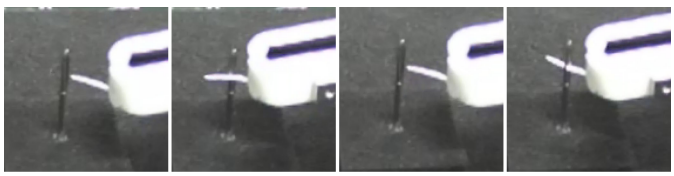


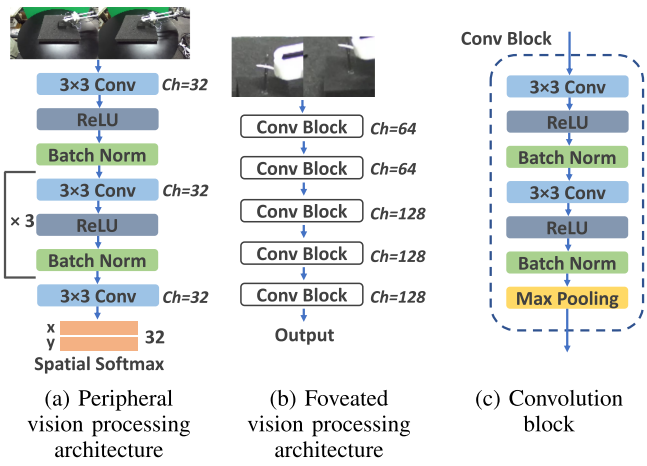
Fig. 4. Proposed architecture.



(a) Approach (b) Failure (c) Recovery (d) Success

Fig. 5. Example of task failure caused by the human operator. The operator failed to thread the needle (5b), recovered from the failure (5c), retried, and finally succeeded (5d).

executes the inferred recovery action for the predicted number of steps. During the recovery steps, the robot continues the recovery action regardless of the change of the *recovery classifier*'s result.



(a) Peripheral vision processing architecture (b) Foveated vision processing architecture (c) Convolution block

Fig. 6. Neural network architectures.

D. Model Architecture

It is important to recognize the three-dimensional state of the thread for the needle threading task. Therefore, both the left and right image of the stereo camera are used as input.

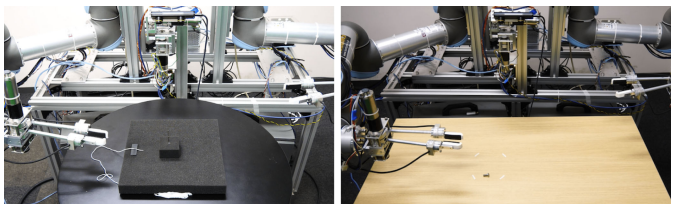
The entire architecture is shown in Fig. 4. First, the raw 1280×720 left and right RGB images are concatenated into six channels and resized to a 128×72 peripheral image. This peripheral image is used to infer the fast-action and left/right gaze coordinate. The left/right foveated images are cropped from left/right 1280×720 raw images using the inferred left/right gaze coordinate, respectively. The left/right foveated images are concatenated into six channels. The foveated image is used to infer (1) the gripper opening/closing, (2) standard slow-action, (3) recovery slow-action, (4) recovery classification, (5) recovery step prediction, and (6) action speed classification (slow/fast) (Fig. 4). The *action speed classifier* determines whether to use the slow-action or fast-action in the current state. The *recovery classifier* predicts whether the robot has failed to thread the needle. If the classifier decides to use the slow-action and recovery is required in the current state, the recovery action is selected and executed for n steps, where n is the number of steps predicted by the *recovery step predictor*. If recovery is not required, a standard (fast or slow) action is executed based on the decision of the *action speed classifier* (Algorithm 1).

Similar to previous research [12], gaze coordinate is inferred by the mixture density network (MDN) [34]. The MDN computes the parameters of the GMM to estimate the probability distribution of the target conditioned on the input data. In this study, the MDN architecture inputs a concatenated left/right peripheral image at the current time step to output μ, σ, and ρ, which represent the mean, standard deviation, and correlation of the two-dimensional gaze position probability, respectively.

The coordinate of the end-effector and the target is important for fast-action control using the peripheral image because the fast-action control moves the end-effector into the vicinity of the target. To extract the coordinate information as a feature, spatial softmax [1], [10], which represents a feature as a two-dimensional coordinate (Fig. 6(a)), is used. The gaze predictor

TABLE I
NEEDLE THREADING RESULT. THE PROPOSED METHOD WAS SIGNIFICANTLY BETTER THAN THE TASK RESULTS MARKED USING* (CHI-SQUARE TEST, $P < 0.05$)

Experiment	Method	Pick (16 trials) (%)	Thread (16 trials) (%)
(a) Assessment of the proposed method	Proposed method	93.75	81.25
(b) Assessment of foveated vision	Without gaze (image size: 128 × 72)	93.75	12.50*
	Without gaze (image size: 1280 × 720)	0.00*	0.00*
(c) Assessment of action separation	Without action separation	43.75*	0.00*
	Action speed threshold = 0.00208	25.00*	12.50*
	Action speed threshold = 0.00364	93.75	68.75
	Action speed threshold = 0.00790	93.75	50.00
	Action speed threshold = 0.00985	87.50	43.75*
(d) Assessment of visuomotor control separation	Fast-action with foveated vision only	0.00*	0.00*
	Fast-action with foveated + peripheral vision	6.25*	0.00*
	Slow-action with foveated + peripheral vision	100.00	62.50
(e) Assessment of recovery action	Standard & recovery action in one network	93.75	50.00
	No *recovery step predictor*	93.75	31.25*
(f) Assessment of resolution	Half resolution (640 × 360)	100.00	50.00
(g) Assessment of stereo vision	Left image only	68.75	12.50*



(a) Needle threading (b) Bolt picking

Fig. 7. Task setup.

also uses the network architecture with the spatial softmax (Fig. 6(a)) because the target location is important in gaze coordinate prediction.

By contrast, the recognition of the three-dimensional state of the thread and the needle is important for the processing of the foveated image. Spatial softmax is inadequate for inferring such information because the extracted feature is a two-dimensional coordinate. Therefore, the foveated image is processed using a series of convolutional neural networks and max-pooling layers with a stride of two (Fig. 6(b), 6(c)).

The extracted features are processed with a series of fully-connected (FC) layers. Batch normalization [35] and the ReLU activation function [36] are used between the FC layers. There are 200 nodes for each FC layer. Four neural network models that infer actions (standard slow-action, recovery slow-action, fast-action, and gripper opening/closing) use three FC layers followed by Fig. 6, whereas the other neural network models use two FC layers.

IV. EXPERIMENTS

A. Task Setup

In the needle threading task (Fig. 7(a)), the robot manipulator had to (1) pick up the thread and (2) insert it into the eye of the needle that was randomly placed on a 9 cm × 9 cm area on the table. In this setup, the trials varied in the location of the needle and the grasping position of the thread (Fig. 2). The neural network had to adjust its policy with such variances.

TABLE II
EFFICIENT LEARNING OF NEURAL NETWORKS USING THE GAZE (NEEDLE THREADING)

Method	MACs (G)	Learning time (hours)
Proposed method	7.21	7.92
Without gaze (1280 × 720)	58.3	220

TABLE III
BOLT PICKING RESULT

Method	Pick (16 trials) (%)
(a) Proposed method	68.75
(b) Without gaze (image size: 1280 × 720)	18.75
(c) Without action separation	0.00

TABLE IV
MEDIAN OF THE GAZE PREDICTION ERROR (HORIZONTAL: $error\_in\_pixels/width \times 100$, VERTICAL: $error\_in\_pixels/height \times 100$)

Dataset	Left (%)		Right (%)	
	Horizontal	Vertical	Horizontal	Vertical
Training set	0.670	1.69	0.639	1.10
Validation set	0.620	1.61	0.672	1.04

To demonstrate that the proposed method is generally applicable to precise tasks, a bolt-picking task was also conducted (Fig. 7(b)). In this task, the robot had to pick up a $M6 \times 14$ bolt. Hence, the robot had to learn the precise location to grasp the bolt while avoiding colliding with the table.

B. Assessment of the Proposed Method

The proposed method achieved a success rate of 81.25% for needle threading using 203.8 minutes of the training data and 68.75% in bolt picking using 39.56 minutes of the training data (Tables I(a) and III(a), respectively). Fig. 8 illustrates that the robot grasped the thread (6.0 s), recognized the failure (16.0 s), recovered from the failure state (18.0 s), and finally succeeded in threading the needle (22.0 s). In another trial (Fig. 9), the robot was able to perform the appropriate manipulation, even when the thread deformed after colliding with the needle. Table IV shows the gaze predictor accuracy.

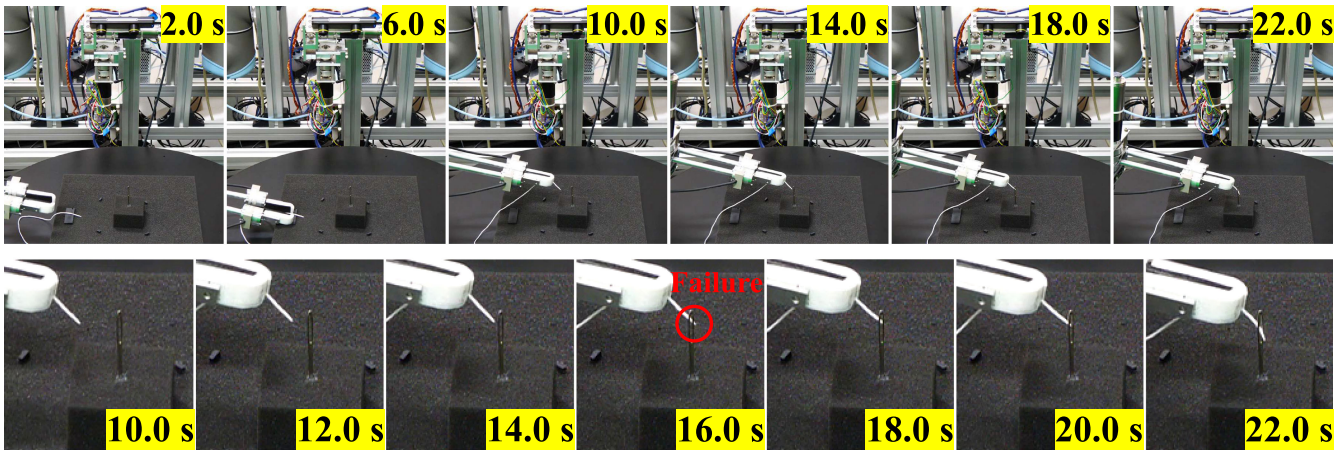


Fig. 8. Example of a successful trial. The robot was able to recognize the failure (16.0 s), recover from it (18.0 s), and finally succeed in threading the needle.

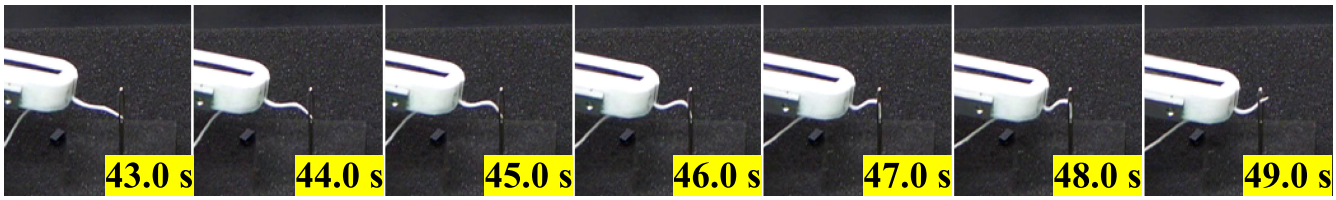


Fig. 9. Another successful trial. The proposed method was able to perform manipulation, even when the thread was deformed.

C. Assessment of Foveated Vision

To evaluate how the gaze affects both computational cost and performance, the proposed method was compared with methods that do not use foveated vision. When the foveated images were replaced with the peripheral image (128×72), the neural networks failed at needle threading (success rate of 12.5%). Moreover, when all images were replaced with the raw image (1280×720), the neural networks even failed at picking a thread (Table I(b)). This network architecture also failed at bolt picking (Table. III(b)). Therefore, selecting an image with an appropriate resolution in accordance with the target task contributes to the improvement of the generalization.

The computational cost of the proposed method and that of processing a raw 1280×720 image were compared (Table II). The proposed method required 8.10 times fewer multiply-accumulate operations (MACs) to calculate one step, and approximately 27.7 times less computational time to train the entire 30 epochs of all neural networks. A difference between the saved computational time and the saved MACs was caused by overheads such as memory copy.

D. Assessment of the Action Separation

The success rate of needle threading dropped to 0% when both the fast-action and slow-action were trained using only the peripheral image. Additionally, the threshold search results confirmed that the threshold that separates slow-action from fast-action should be the intersection point of the two Gaussian distributions (Tables I(c) and III(c)). In addition to the intersection point of $0.005\,685 \sim \mu_1 + 2.315\sigma_1 \sim \mu_2\text{-}1.069\sigma_2$, four possible thresholds were tested: $[0.00\,208, 0.00\,364, 0.00\,790, 0.00\,985]$, which correspond to $[\mu_1, \mu_1 + \sigma_1, \mu_2\text{-}\frac{\sigma_2}{2}, \mu_2]$, respectively. μ_1, μ_2, and σ_1, σ_2 are the mean and standard deviation of each Gaussian ($\mu_1 < \mu_2$), respectively.

E. Assessment of Visuomotor Control Separation

The proposed method assigned peripheral vision to the fast-action and foveated vision to the slow-action. For other architectures, such as allocating only foveated vision to both the fast-action and slow-action or assigning both foveated and peripheral vision to the fast-action, the robot failed to pick up the thread. Moreover, allocating both foveated and peripheral vision to slow-action decreased the success rate (Table I(d)).

F. Assessment of the Recovery Action

Training both standard and recovery actions using only one network resulted in inferior performance (Table I e). The reason for this is that the recovery action was not explicitly acquired, thus the robot could not recover after failure. Notably, the proposed method was able to recover from 12 out of 13 failures, whereas the neural network trained with both standard and recovery actions only recovered from one out of eight failures. Additionally, the success rate dropped to 31.25% without the *recovery step predictor*. At this time, the robot repeatedly switched between the standard action and the recovery action.

G. Assessment of the Resolution

When a half-resolution image (640×360) was used, the success rate dropped to 50% (Table I f), which demonstrates that high-resolution images are required to conduct needle threading tasks.

H. Assessment of Stereo Vision

To evaluate the necessity of stereo vision, the entire neural network architectures were re-trained with mono vision using images from the left camera only (therefore, the number of the input channels was three). Without stereo vision, the success rate dramatically decreased (Table I(g)).

V. Discussion

In this paper, we proposed a gaze-based dual resolution deep imitation learning method for robot manipulation. The proposed method separates the foveated and peripheral vision, which are used to control the slow-action and fast-action, respectively. The proposed method enabled a general-purpose robot manipulator to perform needle threading with high precision. This approach achieved appropriate control even when the thread was deformed. Additionally, the experimental results obtained for the bolt picking task partially demonstrated that the proposed method can be generalized and applied to other tasks that require high precision. As the proposed method enables the high-precision manipulation of the a deformable object only from human demonstration data, it is potentially applicable to tasks that require dexterous manipulation skills such as food processing, jewelry manufacturing, or clothing manipulation.

The results obtained in this study confirm that this separated visuomotor control improves both the task performance and computational efficiency compared with processing using high-resolution visual input. Moreover, the results of this research agree with the results obtained by previous studies on the role of central/peripheral vision in visuomotor manipulation [3], [6], [9]. This suggests that the characteristics of human behavior must be considered in the application of imitation learning using human-generated data to robots, and that only using end-to-end fitting for the motor control output from input sensory data may be inadequate.

In this research, the foveated vision is defined as 10° around the gaze position. The experimental results demonstrated that this setup can solve tasks that handle small objects. Whether the same setup can solve tasks with large objects still requires investigation. The area of the foveated vision can be related to the target objects. Therefore, the active adjustment of the foveated vision area in accordance with the target task may further increase the generality of the proposed method. Additionally, the stereo camera was fixed in this work. To fully use the advantages of active vision, active control of the camera head, in addition to gaze coordinate prediction in the image, is required.

Appendix

A. Calculation of Foveated Vision From the Camera Image

The field of view of the ZED Mini stereo camera is $90^\circ(H) \times 60^\circ(V) \times 100^\circ$. Therefore, the central retina (10°) corresponds to 142×120 of the image from the full image size of 1280×720.

B Training Details

The neural networks were trained on needle threading using 1713 episodes of training data (203.8 minutes) and 148 episodes of validation data (17.01 minutes). For bolt picking, 775 training episodes (39.56 minutes) and 102 validation episodes (5.402 minutes) were used. The demonstration was sampled at 10.0 Hz. Each neural network component was selected from the lowest validation loss over 30 epochs. The learning rate of $1e-4$, rectified Adam (RAdam) optimizer [37], and the same action loss as that in [12] were used.

Each neural network component was trained using a Xeon CPU E5-2698 v4 and an NVIDIA Tesla V100 GPU with a batch size of 64. When training was conducted using a 1280×720 image, a batch size of 8 was used to avoid memory issues. An Intel CPU Core i7-8700 K and one NVIDIA GeForce GTX 1080 Ti were used to control the UR5 robot manipulator.

TABLE V
Classification Accuracy

Classifier	Training set (%)	Validation set (%)
Recovery classifier	97.43	94.90
Action speed classifier	94.20	93.99

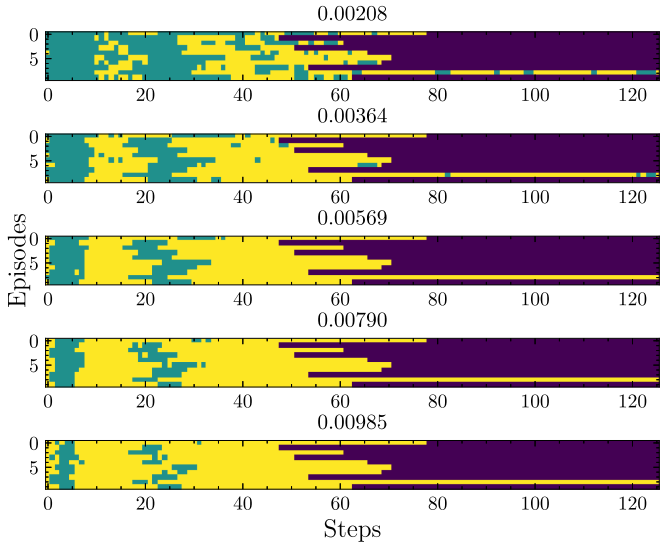


Fig. 10. Fast/slow-action visualized on each threshold (horizontal: steps in an episode, vertical: episodes, cyan: fast-action, *yellow*: slow-action).

C. Evaluation Details

The needle block was a 5 cm $\times$ 5 cm sponge block with the needle installed in the middle, and was placed as similarly as possible to the previously recorded test positions. To ensure a fair comparison of the test results, approximately $1 \sim 2$ mm of the end of the thread was hardened using glue. Notably, this did not result in loss of deformability for the thread. In the additional thread experiment, wherein the thread was not hardened using glue but moistened using a small amount of water, 68.75% of successful threading was recorded. A trial was assessed as a failure when it exceeded the maximum steps of 500.

D. Classifier Accuracy

The classification accuracy of the *recovery classifier* and *action speed classier* are presented in Table V.

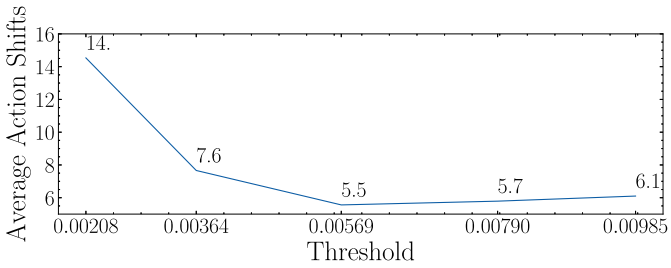


Fig. 11. Average number action shifts (fast→ slow & slow → fast) on each threshold per episode.

E. Assessment of the Action Separation Threshold

Fig. 10 visualizes fast-action (cyan) and slow-action (yellow) separated with thresholds used in IV-D on the validation set of the needle threading task. Threshold $= 0.00\,569$, which is the intersection point of the two Gaussian distributions, showed the least number of action shifts (Fig. 11), which indicates that the proposed method shows the least interleaved sub-segmentation of slow and fast actions.

REFERENCES

[1] T. Zhang *et al.*, "Deep imitation learning for complex manipulation tasks from virtual reality teleoperation," in *Proc. Int. Conf. Robot. Automat.*, 2018, pp. 1–8.

[2] P.-C. Yang, K. Sasaki, K. Suzuki, K. Kase, S. Sugano, and T. Ogata, "Repeatable folding task by humanoid robot worker using deep learning," *Robot. Autom. Lett.*, vol. 2, no. 2, pp. 397–403, 2016.

[3] J. Paillard, "Fast and slow feedback loops for the visual correction of spatial errors in a pointing task: A reappraisal," *Can. J. Physiol. Pharmacol.*, vol. 74, no. 4, pp. 401–417, 1996.

[4] H. Kolb, "Simple anatomy of the retina," Webvision: The Org. Retina Vis. Syst. [Internet]., pp. 13–36, 1995. [Online]. Available: https://www.ncbi.nlm.nih.gov/books/NBK11533/.

[5] M. Hayhoe and D. Ballard, "Eye movements in natural behavior," *Trends Cogn. Sci.*, vol. 9, pp. 188–94, 2005.

[6] A. J. de Brouwer, J. P. Gallivan, and J. R. Flanagan, "Visuomotor feedback gains are modulated by gaze position," *J. Neuriophysiol.*, vol. 120, no. 5, pp. 2522–2531, 2018.

[7] U. Sailer, J. R. Flanagan, and R. S. Johansson, "Eye-hand coordination during learning of a novel visuomotor task," *J. Neurosci.*, vol. 25, no. 39, pp. 8833–8842, 2005.

[8] D. Säfström, R. S. Johansson, and J. R. Flanagan, "Gaze behavior when learning to link sequential action phases in a manual task," *J. Vis.*, vol. 14, no. 4, pp. 3–3, 2014.

[9] F. Sarlegna, J. Blouin, J.-L. Vercher, J.-P. Bresciani, C. Bourdin, and G. M. Gauthier, "Online control of the direction of rapid reaching movements," *Exp. Brain Res.*, vol. 157, no. 4, pp. 468–471, 2004.

[10] C. Finn, X. Y. Tan, Y. Duan, T. Darrell, S. Levine, and P. Abbeel, "Deep spatial autoencoders for visuomotor learning," in *Proc. Int. Conf. Robot. Automat.*, 2016, pp. 512–519.

[11] S. Levine, C. Finn, T. Darrell, and P. Abbeel, "End-to-end training of deep visuomotor policies," *J. Mach. Learn. Res.*, vol. 17, no. 1, pp. 1334–1373, 2016.

[12] H. Kim, Y. Ohmura, and Y. Kuniyoshi, "Using human gaze to improve robustness against irrelevant objects in robot manipulation tasks," *Robot. Autom. Lett.*, vol. 5, no. 3, pp. 4415–4422, 2020.

[13] T. Tang, H.-C. Lin, Y. Zhao, W. Chen, and M. Tomizuka, "Autonomous alignment of peg and hole by force/torque measurement for robotic assembly," in *Proc. Int. Conf. Automat. Sci. Eng.*, 2016, pp. 162–167.

[14] H. Inoue, "Force feedback in precise assembly tasks," *Massachusetts Inst. Technol.*, no. AIM-308, 1974.

[15] I.-W. Kim, D.-J. Lim, and K.-I. Kim, "Active peg-in-hole of chamferless parts using force/moment sensor," in *Proc. Int. Conf. Intell. Robot. Syst.*, vol. 2, 1999, pp. 948–953.

[16] K. Sharma, V. Shirwalkar, and P. K. Pal, "Intelligent and environment-independent peg-in-hole search strategies," in *Proc. Int. Conf. Control, Automat., Robot. Embedded Syst.*, 2013, pp. 1–6.

[17] S. Huang, K. Murakami, Y. Yamakawa, T. Senoo, and M. Ishikawa, "Fast peg-and-hole alignment using visual compliance," in *Proc. Int. Conf. Intell. Robot. Syst.*, 2013, pp. 286–292.

[18] M. Majors and R. Richards, "A neural-network-based flexible assembly controller," in *Proc. Int. Conf. Artif. Neural Netw.*, 1995, pp. 268–273.

[19] T. Inoue, G. De Magistris, A. Munawar, T. Yokoya, and R. Tachibana, "Deep reinforcement learning for high precision assembly tasks," in *Proc. Int. Conf. Intell Robot. Syst.*, 2017, pp. 819–825.

[20] M. Inaba and H. Inoue, "Hand eye coordination in rope handling," *J. Robot. Soc. Jpn.*, vol. 3, no. 6, pp. 538–547, 1985.

[21] J. Silvério and S. Calinon, "A laser-based dual-arm system for precise control of collaborative robots," 2020, *arXiv:2011.01573*.

[22] S. Huang, Y. Yamakawa, T. Senoo, and M. Ishikawa, "Robotic needle threading manipulation based on high-speed motion strategy using high-speed visual feedback," in *Proc. Int. Conf. Intell Robot. Syst.*, 2015, pp. 4041–4046.

[23] J. Aloimonos, I. Weiss, and A. Bandyopadhyay, "Active vision," *Int. J. Comput. Vis.*, vol. 1, no. 4, pp. 333–356, 1988.

[24] R. Bajcsy, "Active perception," *Proc. IEEE Proc. IRE*, vol. 76, no. 8, pp. 966–1005, Aug. 1988.

[25] D. H. Ballard, "Animate vision," *Artif. Intell.*, vol. 48, no. 1, pp. 57–86, 1991.

[26] J. Soong and C. Brown, "Inverse kinematics and gaze stabilization for the rochester robot head," Univ. Rochester, Tech. Rep., 1991.

[27] J. L. Crowley, P. Bobet, and M. Mesrabi, "Gaze control for a binocular camera head," in *Proc. Eur. Conf. Comput. Vis.*, 1992, pp. 588–596.

[28] G. Sandini and V. Tagliasco, "An anthropomorphic retina-like structure for scene analysis," *Comput. Graph. Image Process.*, vol. 14, no. 4, pp. 365–372, 1980.

[29] Y. Kuniyoshi, N. Kita, T. Suehiro, and S. Rougeaux, "Active stereo vision system with foveated wide angle lenses," in *Proc. Asian Conf. Comput. Vis.*, 1995, pp. 191–200.

[30] K. Kuniyoshi, N. Kita, K. Sugimoto, S. Nakamura, and T. Suehiro, "A foveated wide angle lens for active vision," in *Proc. Int. Conf. Robot. Automat.*, vol. 3, 1995, pp. 2982–2988.

[31] S. D. Whitehead and D. H. Ballard, "Learning to perceive and act by trial and error," *Mach. Learn.*, vol. 7, no. 1, pp. 45–83, 1991.

[32] R. Zhang, Z. Liu, L. Zhang, J. A. Whritner, K. S. Muller, M. M. Hayhoe, and D. H. Ballard, "Agil: Learning attention from human for visuomotor tasks," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 663–679.

[33] D. Chotrov, Z. Uzunova, Y. Yordanov, and S. Maleshkov, "Mixed-reality spatial configuration with a zed mini stereoscopic camera," in *Conf. Techno. Edu. Smart World*, 11, 2018.

[34] C. M. Bishop, "Mixture density networks," Neural Comput. Res. Group, Aston Univ., Tech. Rep., 1994.

[35] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 448–456.

[36] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Proc. Int. Conf. Artif. Intell. Statist.*, 2011, pp. 315–323.

[37] L. Liu *et al.*, "On the variance of the adaptive learning rate and beyond," in *Proc. Int. Conf. Learn. Representations*, 2020, pp.1–14.