Using Human Gaze to Improve Robustness Against Irrelevant Objects in Robot Manipulation Tasks

Heecheol Kim, Yoshiyuki Ohmura, and Yasuo Kuniyoshi

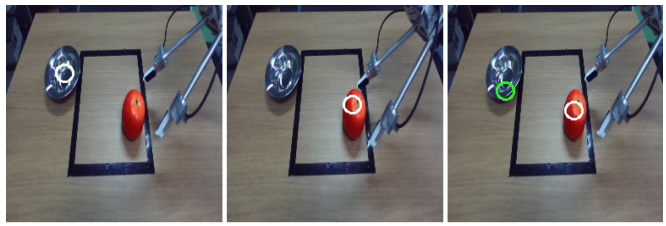
***Abstract*—Deep imitation learning enables the learning of complex visuomotor skills from raw pixel inputs. However, this approach suffers from the problem of overfitting to the training images. The neural network can easily be distracted by task-irrelevant objects. In this letter, we use the human gaze measured by a head-mounted eye tracking device to discard task-irrelevant visual distractions. We propose a mixture density network-based behavior cloning method that learns to imitate the human gaze. The model predicts gaze positions from raw pixel images and crops images around the predicted gazes. Only these cropped images are used to compute the output action. This cropping procedure can remove visual distractions because the gaze is rarely fixated on task-irrelevant objects. This robustness against irrelevant objects can improve the manipulation performance of robots in scenarios where task-irrelevant objects are present. We evaluated our model on four manipulation tasks designed to test the robustness of the model to irrelevant objects. The results indicate that the proposed model can predict the locations of task-relevant objects from gaze positions, is robust to task-irrelevant objects, and exhibits impressive manipulation performance especially in multi-object handling.**

***Index Terms*—Deep learning in grasping and manipulation, learning from demonstration, telerobotics and teleoperation, visual servoing, computer vision for automation.**

I. Introduction

IMITATION learning involves learning a policy by observing expert demonstrations. One application of imitation learning is in robotics (e.g., [1]–[4]), because this method offers potential for learning complex policies. Imitation learning does not require further exploration, unlike reinforcement learning, which requires implausible amounts of interactions to train robots [5]. By using teleoperation systems, high-quality demonstration data for manipulation tasks become easily available [6].

Deep learning has been used to solve high-dimensional computer vision tasks, such as classification, object detection, and semantic segmentation (e.g., [7]–[10]). Learning robot manipulation from raw pixel images has been studied because there is no need to manually define the state space from high-dimensional image inputs (e.g., [11]–[15]). However, direct mapping from the images to the action output suffers from overfitting to the training images. In case there are changes in the background (i.e., the advent of task-irrelevant objects), they change the network's policy output. This is because the mapping of the output action from visual features relies on fully connected layers. Changes of features due to the advent of objects affect the output of matrix multiplication in the fully connected layers.



(a) Ground truth human gaze (b) Ground truth human gaze (0.4s after Fig. 1a) (c) Gazes predicted by MDN

Fig. 1. MDN architecture can infer multiple gazes (1c) from a single gaze (Fig. 1a, 1b) by a human operator.

Human gazes at task-relevant object positions [16], [17]. Therefore, we can remove information about task-irrelevant objects by measuring the gaze of a human operator using a head-mounted eye tracking device while teleoperating with a robot. The acquired gaze position as well as state-action demonstration pairs are used to learn manipulation tasks. As this method discards out-of-gaze objects to change the policy, the policy is robust to such visual distractions as the advent of unseen and new objects. Robustness against visual distractions also improves generalization on tasks involving multiple objects. In such tasks, all other objects that are irrelevant to the given manipulation sub-task (i.e., all objects that are not manipulated, where usually one target can be manipulated at one time) become visual distractions.

People can gaze at different positions in similar situations; i.e., when completing a task involving two objects, a person can gaze at one object at time step t and the other at time step $t+1$. In this case, gaze labels are assigned to different objects on very similar input images (e.g., Fig. 1a and 1b). When such prediction of the gaze is considered to be object localization (i.e., regression), the problem is intractable because the object regression model

Manuscript received December 21, 2019; accepted May 12, 2020. Date of publication May 28, 2020; date of current version June 9, 2020. This letter was recommended for publication by Associate Editor Lorenzo Jamone and Editor Tamim Asfour upon evaluation of the reviewers' comments. 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 113-8656, 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 article has supplementary downloadable material available at https://ieeexplore.ieee.org, provided by the authors.

Digital Object Identifier 10.1109/LRA.2020.2998410

tries to fit one output coordinate to two gaze positions from two similar input images. Therefore, we consider gaze position prediction to be a problem of probability estimation. The Mixture Density Network (MDN) [18] estimates parameters of the Gaussian Mixture Model (GMM) from the given target and can reconstruct the probability distribution of gaze positions. Thus, the MDN can infer multiple object locations simultaneously, even though it is trained using only one gaze point at each time step (Fig. 1 c).

The main contributions of this paper are as follows:

1) To the best of our knowledge, this research is the first to use the human gaze to improve imitation learning performance for robot manipulation tasks.
2) We propose using the MDN to predict the human gaze.
3) We empirically show that gaze prediction makes the learning policy more robust to visual distractions and improves multi-object manipulation performance.

II. Related Work

Imitation learning includes inverse reinforcement learning and behavior cloning. Inverse reinforcement learns the reward function from the trajectories of demonstrations [19]–[22]. Adversarial optimization [23] has been recently applied to inverse reinforcement learning [24], [25]. However, these adversarial training-based methods require a large number of interactions with the environment, which is unfeasible when training robots.

Behavior cloning involves learning direct mapping from the input state to the output action using the given expert demonstration. The successful applications of behavior cloning to robotics include [6], [13], [26], [27]. In [6], a teleoperation system integrated with virtual reality is proposed to make natural human-robot operating interface and showed that end-to-end learning of neural network by behavior cloning can accomplish manipulation tasks. However, no research was done for applying visual attention mechanism for robustness to task-irrelevant objects.

Predicting the location at which the gaze is fixed (i.e., saliency map) is an important task in computer vision that aims to describe visual attention. Many recent studies have investigated predicting the saliency map [28]–[33]. Even though impressive results have been reported, they are not applicable to robot control, because the saliency map is high dimensional. Our proposed method does not output the saliency map directly but generates low-dimensional mean gaze positions therefore its output can easily guide policy.

Eye tracking has been studied in robotics for many purposes, such as utilizing gaze information as a control signal to robots [34]–[36] and human–robot communication [37], [38]. To the best of our knowledge, no study has addressed improving the imitation learning performance of robots by using gaze information.

There is a research to use gaze prediction for imitation learning in Atari games [39]. But our research goal is to address improving the manipulation performance of real robots by using human gaze information. In such a situation, the human gaze is highly correlated to handling objects [17], and human rarely gazes at task-irrelevant positions [16]. This correlation between gaze and task-relevant objects grants validity to our approach to discard visual information that is not the focus of the gaze. But, in Atari games, such correlation is currently unknown.

III. Robot System With Eye Tracking

A. Hardware

Our robot teleoperation system was composed of two UR5 robot arms, a stereo camera on the robot, two robot controllers, and a head-mounted display with an eye tracker on it. To generate demonstration data, a human operator operated the UR5 robots through robot controllers while wearing a head-mounted display (HMD) that showed images from the stereo camera.

The ZED Mini stereo camera developed by Stereo Labs was mounted between the UR5 robot arms, providing stereo RGB images to the operator. The frame rate of this stereo camera was 35 Hz, sufficiently smooth for precise operation. A Tobii Pro VR Integration is the integration of an HTC Vive HMD and a Tobii Pro eye tracking device. The Tobii Pro VR Integration provides stereo video images to the human operator and measures the gazes of both the left and right eyes simultaneously.

We built two controllers that were kinematically equivalent to two UR5 robots developed by Universal Robots. The controllers were implemented with $6 + 1$ (DoF of UR5 robot+DoF of gripper) encoders that measured the angles of the joints of each DoF. The UR5 robots were operated at 100 Hz with the measured joint angle signal. The human operator could operate the robots naturally and accurately with these controllers. In the experiments, only the right UR5 robot and controller were used.

The neural networks were trained using Intel Xeon CPU E5-2698 v4 and single NVIDIA Tesla V100. Intel CPU Core i7-8700K was used for inference.

B. Data Processing

We measured gaze information when the human operator conducted all tasks. The measured time-series of human gaze is used to train the MDN model. On test, gaze is inferenced from camera input by using the trained model.

The RGB image was resized to $256 \times 256 \times 3$ to train the proposed neural network model. Between the left and right camera images, the side of the operator's dominant eye (throughout this research, left) was chosen. Although the ZED Mini can generate a depth image from stereo images, we did not use it because our experiments which were always performed on the same flat table do not require 3D information.

The angles of the joints of the right UR5 robot were converted into the position of the end-effector $[x, y, z]$ and orientation represented by the Euler angle $[\alpha, \beta, \gamma]$. The angles of the gripper were binarized into $g_t \in \{0, 1\}$ which represented the opening/closing of the gripper. The position and orientation of the end-effector represented by the Euler angle were concatenated in $p_t \in \mathbb{R}^6$. $s_t \in \mathbb{R}^7$ is the concatenation of p_t and g_t, and represents the state of the robot's arm including the gripper.

The input to the neural network model was $I_t = (o_t, s_{t-4:t})$, where $o_t \in \mathbb{R}^{256\times256\times3}$ was the resized RGB image and $s_{t-4:t} \in$

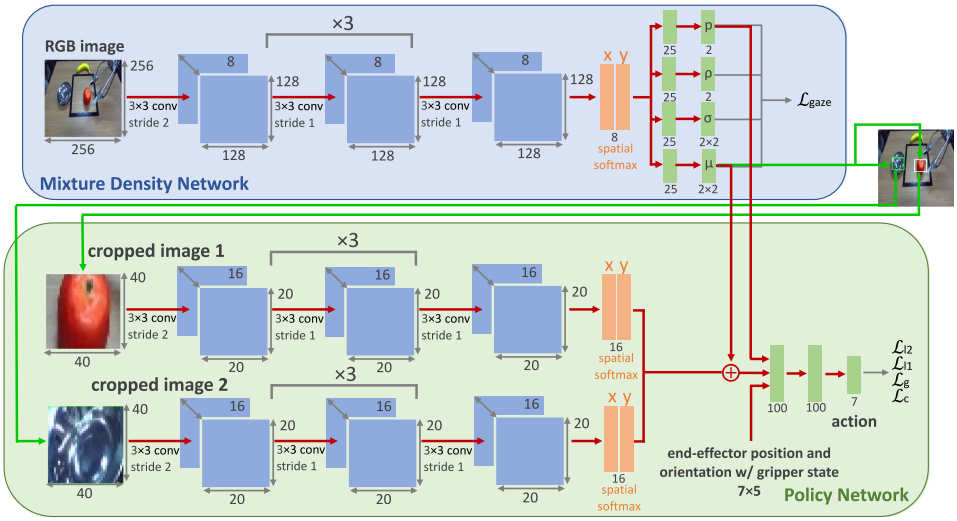


Fig. 2. Architecture of our neural network model. The mixture density network (blue) predicts gazes from an input RGB image, and 40×40 cropped images around the predicted gaze positions are used to predict the action output of the next step (green).

$\mathbb{R}^{7\times5}$ were the five most recent steps of the states of the end-effector (see [6]). The target $O_t = (u_t, g_{t+1}, e_t)$ included (1) the end-effector's action command $u_t = p_{t+1} - p_t \in \mathbb{R}^6$, (2) the gripper's command $g_{t+1} \in \mathbb{R}^1$, and (3) the dominant gaze position $e_t \in \mathbb{R}^2$.

IV. Behavior Cloning With Gaze Prediction

The proposed model aims to learn the mapping from input pixels and states for action outputs through behavior cloning. Unlike conventional behavior cloning, the proposed model does not directly map the input image to the output. It first trains the MDN to learn the gaze positions, and cropped images around the predicted gaze positions are directly mapped to the output action. The proposed model architecture is based on the baseline method [6] which studied the behavior cloning of manipulation tasks using teleoperation demonstration data (see Appendix B for details).

A. Mixture Density Network

The MDN [18] is a neural network that learns the probability distribution of given target data from the input by assuming that the data probability is a mixture of Gaussian distributions. To focus on learning not temporal but spatial property of gazes, we used MDN without any recurrent structure. As the output of the MDN can generate multiple gazes, unlike humans, the gazes directed at two objects can be generated simultaneously while the human needs to temporarily switch gaze between task-relevant objects.

Without the recurrent layer, the outputs of the MDN $\mu_t^n, \sigma_t^n, \rho_t^n$, and p_t^n depend only on the given input image o_t:

$$\mu_t^n = \pi_\mu(o_t) \tag{1}$$

$$\sigma_t^n = exp(\pi_\sigma(o_t)) \tag{2}$$

$$\rho_t^n = tanh(\pi_\rho(o_t)) \tag{3}$$

$$p_t^n = softmax(\pi_p(o_t)) \tag{4}$$

where μ_t^n, σ_t^n, and ρ_t^n refer to the mean, standard deviation, and correlation of 2D gazes of the nth Gaussian distribution at time step t, and p_t^n indicates the weight of the nth Gaussian distribution. π_μ, π_σ, π_ρ, and π_p refer to neural network architecture that computes μ, σ, ρ, and p respectively. The probability of the given gaze position e_t under the MDN model is estimated by $\mathcal{N}(e_t; \mu_t^n, \sigma_t^{n2}, \rho_t^n)$, and the weighted sum of probabilities with weight p_t^n is maximized by using the negative log-likelihood loss:

$$\mathcal{L}_{gaze} = -log\left(\sum_{i=1}^{N} p_t^n \mathcal{N}(e_t; \mu_t^n, \sigma_t^{n2}, \rho_t^n)\right) \tag{5}$$

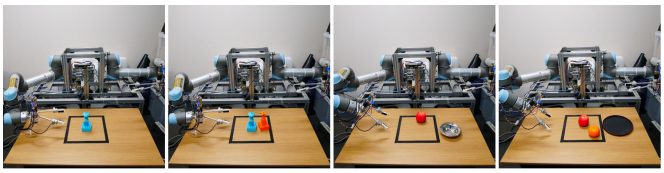
B. Model Architecture

The architecture of our neural network is illustrated in Fig. 2. The input image is convolved using five convolutional layers [7] with eight channels, and a spatial softmax layer [11], [12] is used to compute spatial feature points from pixels (see [12] for detail). These points are then passed to two fully connected layers to learn $\mu_t^n, \sigma_t^n, \rho_t^n$, and p_t^n. we used two Gaussian distributions.

The policy network learns visuomotor policy from cropped images around the predicted gaze positions. We cropped a 40×40 area around the predicted gaze position. This network was composed of two sub-modules of five convolutional layers with spatial softmax and three fully connected layers. The output of spatial softmax was added to the gaze position μ to acquire the absolute position of the feature point on the image. The concatenation of the five most recent end-effector states $s_{t-4:t} \in \mathbb{R}^{7\times5}$, probability of each Gaussian p_t^n, and the output of spatial softmax of all sub-modules constituted the input to the three fully connected layers that output the action.

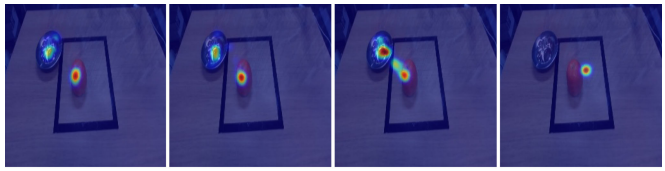
C. Loss Function

The loss function of the policy network closely follows [6]. ℓ_1 loss, ℓ_2 loss, and directional alignment loss fit the output $\pi_\theta(o_t, s_t)$ into the end-effector's action command u_t and the



(a) Pick (b) Pick-2 (c) Pick-and-place (d) Pick-and-place×2

Fig. 3. Task setups.



(a) Ground truth (b) MDN (c) Regression-MLP (d) Regression-LSTM

Fig. 4. Sampled gaze saliency map for each method. Regression-MLP (Fig. 4c) spreads its predictions between objects at which the human operator had mainly gazed.

gripper action command g_{t+1} for the next step. $\mathcal{L}_c$ encourages learning directional alignment rather than the velocity of the action because the latter does not affect performance in our task setup:

$$\mathcal{L}_{\ell 2} = ||\pi_\theta(o_t, s_t) - u_t||_2^2 \tag{6}$$

$$\mathcal{L}_{\ell 1} = ||\pi_\theta(o_t, s_t) - u_t||_1 \tag{7}$$

$$\mathcal{L}_c = arccos\left(\frac{u_t^T \pi_\theta(o_t, s_t)}{||u_t||\,||\pi_\theta(o_t, s_t)||}\right) \tag{8}$$

Binary cross-entropy loss was used to predict the gripper's open/close status at the next time step $g_{t+1} \in \{0, 1\}$ with the network's gripper output $\pi_g(o_t, s_t)$:

$$\mathcal{L}_g = g_{t+1} \log \pi_g(o_t, s_t) + (1 - g_{t+1}) \log\left(1 - \pi_g(o_t, s_t)\right) \tag{9}$$

MDN loss $\mathcal{L}_{gaze}$ was calculated as in (5). The weighted sum of loss functions described above was used as the overall loss function to update the network:

$$\mathcal{L}_{total} = \lambda_{\ell 2}\mathcal{L}_{\ell 2} + \lambda_{\ell 1}\mathcal{L}_{\ell 1} + \lambda_c \mathcal{L}_c + \lambda_g \mathcal{L}_g + \lambda_{gaze}\mathcal{L}_{gaze} \tag{10}$$

The gradients of the policy loss functions ($\mathcal{L}_{\ell 2}$, $\mathcal{L}_{\ell 1}$, and $\mathcal{L}_c$, $\mathcal{L}_g$) are not backpropagated to the MDN module because we want the MDN to not be affected by policy updates.

V. Experiments

A. Experimental Setup

Our experiments were designed to evaluate the ability of the proposed model to learn robotic manipulation tasks as well as its robustness against visual distractions.[1] The manipulation tasks were as follows: (1) Pick: picking a LEGO structure (Fig. 3a); (2) Pick-2: picking a LEGO structure in the presence of another LEGO structure (Fig. 3b); (3) Pick-and-place: picking a toy apple and placing it in a bowl (Fig. 3c); (4) Pick-and-place$\times N$: picking and placing N multiple objects in order (Fig. 3d). In Pick-and-place, the bowl was placed along the line on the left side of the 25 cm $\times$ 25 cm area where the apple is placed on (Fig. 3c).

(1) and (2) are designed to evaluate whether the proposed method is robust to the appearance of a new object or the absence of a known object. (3) is chosen to confirm the proposed method is generally applicable to tasks that both picking and placing locations are arbitrary. The purpose of (4) is to evaluate manipulation performance on multi-object tasks. We tested on $N = 2$ (apple and orange) and $N = 3$ (apple, orange, and kiwi). Details of task setups are provided in Appendix A.

[1]The supplementary video that describes the example behavior of the proposed model on each task is available at http://ieeexplore.ieee.org. This video is 48.5 MB in size.

B. Assessment of Performance in Terms of Predicting Gaze

Another option for gaze prediction is to treat it as a regression problem. The output of the model is a single gaze coordinate at each time step in this case. We propose two regression models for comparison: the feed-forward model (Regression-MLP) and the recurrent model (Regression-LSTM). Regression-MLP learns the gaze position from an image at a single time step, and Regression-LSTM considers images of past steps. Both models had the same convolutional layers and a spatial softmax layer with our MDN. Two fully connected layers followed the spatial softmax layer in Regression-MLP, whereas an LSTM and a fully connected layer followed the spatial softmax layer in Regression-LSTM.

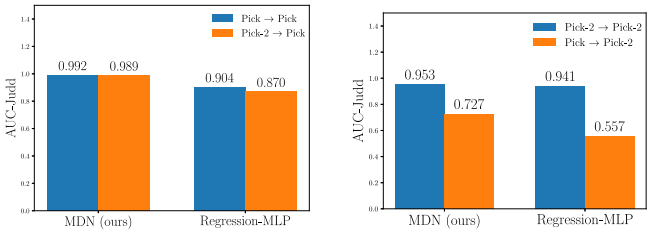
Because the MDN learns multiple gazes from a single ground-truth gaze, simply calculating the distance between the predicted gaze and the ground truth was not appropriate.

Instead, we generated a saliency map for each demonstration of the test set by concatenating all gazes in a demonstration and considering them as a set of gazes for a single image. The saliency map was computed by first setting the values of the pixels at μ_t^n to p_t^n and applying Gaussian blur. p_t^n is the weight of each Gaussian distribution directly generated from the output of the MDN, or always $p_t^n = 1$ for Regression-MLP and Regression-LSTM. Fig. 4 shows how each model predicted the saliency map. Regression-LSTM failed to learn and Regression-MLP often predicted the locations between task-relevant objects. The MDN accurately imitated the ground truth.

We evaluated models to predict saliency maps using the following multiple metrics because there is no established metric [40], [41]: the area under the ROC curve proposed by Judd (AUC-Judd) [42], Pearson's correlation coefficient (CC), normalized scanpath saliency (NSS) that computes the average of the fixation locations along a subject's scanpath [43], similarity (SIM) which is the sum of the minimum values at each point in the distributions that are scaled to sum to one [42], and the Kullback–Leibler divergence (KL) [44]. The results indicate that the MDN had the highest accuracy over all tasks (Table I and Table VII).

TABLE I
AVERAGE AUC-JUDD SCORES OF GAZE PREDICTION METHODS

	MDN	Regression-MLP	Regression-LSTM
Pick	**0.992**	0.989	0.629
Pick-2	**0.953**	0.941	0.592
Pick-and-place	**0.948**	0.937	0.565
Pick-and-place×2	**0.957**	0.931	0.608
Pick-and-place×3	**0.960**	0.950	0.589



(a) Evaluation with Pick (b) Evaluation with Pick-2

Fig. 5. Comparison of the methods in terms of the AUC-Judd score on Pick and Pick-2. MDN delivered better performance than Regression-MLP, and the Pick → Pick-2 prediction showed a more noticeable drop in score than Pick-2 → Pick prediction.

TABLE II
TASK SUCCESS RATE OF MODELS TRAINED WITH PICK. TASK "PICK" INDICATES THAT MODELS WERE TRAINED ON THE PICK DATASET AND TESTED ON THE PICK SETUP (PICK → PICK). TASK "PICK-2" INDICATES THAT THE MODELS WERE TESTED ON THE PICK-2 SETUP (PICK → PICK-2). THE RESULTS ARE REPRESENTED IN TERMS OF MEAN (STANDARD DEVIATION)

	Task	Mean ($N = 36$)	Best ($N = 9$)
Baseline	Pick	94.4 (6.42) %	88.9 %
	Pick-2	22.2 (25.7) %	44.4 %
Proposed	Pick	**97.2 (5.56) %**	**100 %**
	Pick-2	**44.4 (52.1) %**	**100 %**

Fig. 5 presents a comparison of the methods in terms of AUC-Judd scores between tasks with visual distractions and tasks without visual distractions. Fig. 5b shows the scores on the Pick-2 → Pick-2 test (trained with the Pick-2 training set and tested on the Pick-2 test set, blue tower block) and Pick → Pick-2 test (trained with the Pick training set and tested on the Pick-2 test set, orange tower block). Fig. 5a shows the scores on the Pick → Pick test and Pick-2 → Pick. The results imply that (1) MDN outperformed regression methods, and (2) predicting the gaze position in Pick-2 using the model trained on Pick was more difficult than predicting the gaze position in Pick using the model trained on Pick-2.

C. Evaluating Performance on Manipulation Tasks

In this subsection, the comparison between the baseline and the proposed method was conducted. We trained and tested both models with four random seeds $[0, 1, 2, 3]$. For each random seed, 9 test trials have been performed. Overall success rates of the proposed method were significantly better than the baseline ($p < 1.74\text{e}{-29}$, the chi-squared test. Tables II, III, IV, and V).

Tables II, III, and IV investigate the robustness of models against visual distractions. Table II shows that the models were trained on the Pick dataset and evaluated on the Pick setup (Pick → Pick) or the Pick-2 setup (Pick → Pick-2). In Pick → Pick-2, the models needed to adapt to the appearance of an unseen object. The models listed in Table III were trained on the Pick-2 dataset. The proposed model showed higher success rate than baseline on Pick → Pick (44.4% versus 22.2%) and Pick-2 → Pick (61.1% versus 47.2%). Also, the comparison between Pick → Pick and Pick → Pick-2 showed steep drop of success rate (97.2% → 44.4%, 52.8% drop, proposed method), while comparison between Pick-2 → Pick-2 and Pick-2 → Pick showed more moderate drop of success rate (83.3% → 61.1%, 22.2% drop, proposed method) than models trained with Pick. This implies that (1) the proposed model was in general more robust to visual distractions because it prevented them from changing the policy, and (2) the trained visuomotor policies had poorer adaptation ability to visual distraction caused by the appearance of new objects than that caused by the absence of objects. Table IV presents the success rate on Pick-and-place. The models were tested with (1) the Pick-and-place setup and (2) the Pick-and-place-with-Banana setup, where a toy banana was placed at a fixed location as a source of visual distraction. The proposed method recorded higher mean success rate than

TABLE III
TASK SUCCESS RATE OF MODELS TRAINED ON PICK-2. TASK "PICK-2" INDICATES THAT THE MODELS WERE TRAINED ON THE PICK-2 DATASET AND TESTED ON THE PICK-2 SETUP (PICK-2 → PICK-2). TASK "PICK" INDICATES THAT THE MODELS WERE TESTED ON THE PICK SETUP (PICK-2 → PICK)

	Task	Mean ($N = 36$)	Best ($N = 9$)
Baseline	Pick-2	**91.7 (10.6) %**	100 %
	Pick	47.2 (42.9) %	**88.9 %**
Proposed	Pick-2	83.3 (14.3) %	100 %
	Pick	**61.1 (21.3) %**	77.8%

TABLE IV
TASK SUCCESS RATE OF MODELS TRAINED ON PICK-AND-PLACE. THIS SHOWS THE SUCCESS RATES OF "PICKING" AND "PLACING" ON THE PICK-AND-PLACE SETUP. TASKS "PICK W/ BANANA" AND "PLACE W/ BANANA" REFER TO THE PICK-AND-PLACE SETUP WITH A TOY BANANA PLACED AS A NEW OBJECT (SEE FIG. 6)

	Task	Mean ($N = 36$)	Best ($N = 9$)
Baseline	Pick	86.1 (16.7)%	77.8%
	Place	83.3 (19.2)%	66.7%
	Pick w/ Banana	41.7 (48.3)%	88.9%
	Place w/ Banana	33.3 (39.5)%	**77.8%**
Proposed	Pick	**94.4 (6.42)%**	**100%**
	Place	83.3 (6.42)%	**77.8%**
	Pick w/ Banana	**58.3 (41.9)%**	88.9%
	Place w/ Banana	**38.9 (33.3)%**	66.7%

TABLE V
TASK SUCCESS RATE OF MODELS TRAINED WITH PICK-AND-PLACE×2 AND PICK-AND-PLACE×3. {APPLE, ORANGE, KIWI} REFER TO THE SUCCESS RATE OF PICK-AND-PLACING EACH OBJECT ON A PLATE

	N	Sub-task	Mean ($N = 36$)	Best ($N = 9$)
Baseline	2	Apple	8.33 (5.56)%	11.1%
		Orange	0.00 (0.00)%	0.00%
	3	Apple	41.7 (22.9)%	66.7%
		Orange	11.1 (22.2)%	44.4%
		Kiwi	0.00 (0.00)%	0.00%
Proposed	2	Apple	**91.7 (22.9)%**	**100%**
		Orange	**72.2 (42.1)%**	**100%**
	3	Apple	**97.2 (5.56)%**	**100%**
		Orange	**88.9 (12.8)%**	**100%**
		Kiwi	**66.7 (28.7)%**	**100%**

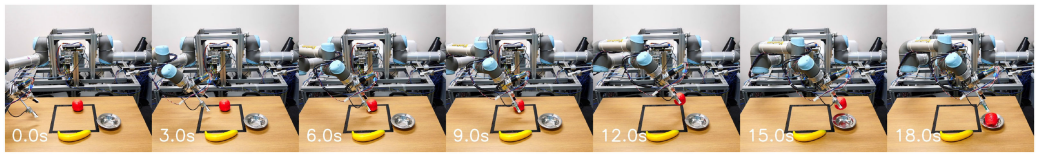


Fig. 6. Example of successful trial of Pick-and-place w/ Banana.

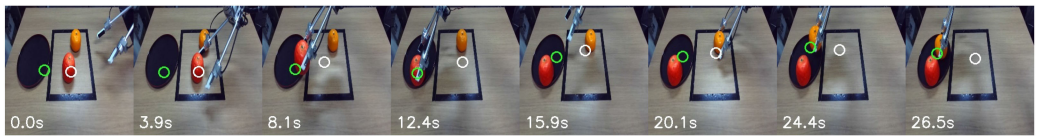


Fig. 7. Example of successful trial of Pick-and-place×2 with the gazes plotted.

baseline on Pick-and-Place → Pick-and-Place w/ Banana. Fig. 6 depicts a successful trial example of Pick-and-Place w/ Banana.

The robustness against irrelevant objects improved manipulation performance on tasks involving multiple objects (Table V). On tasks with multiple objects, the objects became task-irrelevant visual distractions when they were not being manipulated. For example, in Pick-and-place×2, the orange was a visual distraction while picking and placing the apple. Let us assume that the baseline was overfitted to the demonstrations, including in a situation when the apple was placed at position (x_1, y_1) and the orange at (x_2, y_2). In the test phase, when the apple was placed in the same position (x_1, y_1) but the orange was placed at a different position (x_3, y_3), the orange became the visual distraction. Thus, the baseline model could not generalize to manipulate the apple. To deal with this problem without gaze prediction, a very large number of demonstrations are required because the state space to learn expands exponentially according to the number of objects. Cropping images by the predicted gaze positions restricts information about task-irrelevant objects so that a training policy with a feasible number of demonstrations is possible. Fig. 7 shows that the task-irrelevant object (orange) was not gazed at while manipulating the apple (0 s∼6.0 s), and the gaze shifted to the orange as soon as manipulation of the apple was finished (10.4 s∼22.9 s).

VI. Discussion

In this paper, we proposed the use of eye tracking to improve imitation learning by robots to perform manipulation tasks. An MDN-based architecture was proposed to learn visual attention and crop images around the predicted gazes to prevent a degradation in performance owing to visual distractions. The MDN exhibited higher accuracy of gaze prediction than other implementations, including Regression-MLP and Regression-LSTM, because the proposed method learns the distribution of gazes using a GMM rather than simply considering gaze prediction to be a regression problem. The experimental results suggest that the proposed model is more robust to irrelevant objects and provides higher performance, especially on a multi-objects manipulation task.

The results imply that the convolutional layers with a spatial softmax layer, which was used in both the baseline and the proposed model, was less affected by the absence of objects that were always seen during training than the appearance of task-irrelevant objects. One plausible explanation for this is that the disappearance of objects merely rendered the values of the corresponding feature map similar to the feature values of the background (e.g., in Pick and Pick-2, the texture of the table). Because the spatial softmax layer learned task-relevant locations in feature map and ignored task-irrelevant known features, the absence of the task-irrelevant objects had little impact on the output of this layer. On the contrary, because the feature values of unknown objects could not be removed completely due to the infinite variability, the spatial softmax layer was relatively affected by such unknown features. Consequently, our method was more effective when task-irrelevant objects were constructed by known features.

The main reason for most of the failure cases of the proposed model was due to the failure of gaze prediction. Therefore, improving gaze prediction may further improve the task success rate. Further research on neural network design that is robust to visual distractions is needed.

This research did not investigate the effect of the dynamics of the human gaze on robot manipulation performance. While we were mainly concerned with predicting the spatial properties of gazes, a temporal shift in them might provide important clues when solving more complex tasks. We defer this consideration to future research.

Appendix

A. Task specification

- Pick: The purpose of this simple task was to pick a target object. The target object was randomly placed in a 25 cm × 25 cm area on the table. A successful trial was defined by the robot picking the pillar or the top of the target object with its fingertips such that it was lifted up from the table.
- Pick-2: This task setup was identical to Pick task except that the orange tower block was placed to the left of the target object. The orange tower block could be placed outside the 25 cm × 25 cm area it remained in the vicinity of the target object.
- Pick-and-place: In this task, a toy apple was placed randomly in the 25 cm × 25 cm area on the table, and a bowl

TABLE VI
TRAINING SET STATISTICS WITH THE NUMBER OF DEMONSTRATIONS AND TOTAL DEMONSTRATION TIME

Task	# of demo	Total demo time (min)
Pick	214	8.75
Pick2	180	7.79
Pick-and-place	417	31.9
Pick-and-place×2	480	62.8
Pick-and-place×3	500	88.2

was placed to the left of the area. A trial was considered successful if the robot picked up the apple from the table and dropped it into the bowl.

- Pick-and-place×2: A toy apple was placed in the bottom half of the 25 cm × 25 cm area and an orange in its upper half. All demonstrations first involved placing the apple on the plate and then placing the orange on it. A successful trial was defined by the robot properly picking and placing both the apple and the orange on the plate.
- Pick-and-place×3: The toy apple was placed under the toy orange and the toy orange was placed under the toy kiwi. Other setups follow Pick-and-place×2.

The total dataset is divided into 90 % training set and 10 % validation set. Table VI presents the statistics of the training set of tasks. The demonstration was downsampled from 35 Hz to 5 Hz.

B. Model specifications

In our experiments, $\lambda_{\ell 1} = 0.1$, $\lambda_{\ell 2} = 1.0$, $\lambda_c = 0.5$, and $\lambda_{gaze} = 1.0$ were used with a learning rate of 0.0001 and batch size of 36. We used the Rectified Adam (RAdam) [45] optimizer, a variant of the Adam optimizer.

We excluded the auxiliary prediction network described on [6] from the baseline model for a fair comparison with the proposed model. We adjusted the number of channels of the baseline so that the total number of model parameters was similar to the number of parameters of the proposed model (approx. 46,000). The number of channels of the last convolutional layer was 32 so that the total number of extracted features was identical to that of the proposed model. The last three fully connected layers had the same number of parameters.

For Pick-and-place×3, we used MDN with 16 channels at convolutional layers because the visual input contains more information in this task. The total number of model parameters was 59,500 and the size of the baseline network was adjusted to follow the number of the parameters.

C. Evaluation procedure

When evaluating the models on a robot, the initial positions of the objects were determined by participants who were not aware of the model being tested. The participants were shown samples of the initial images of each demonstration on the test set and were asked to place them identically. Out of the four models with random seeds, the one with the highest total success rate of all tasks was selected as the best. Note that the success rate of each task of the selected model is not always the best.

TABLE VII
GAZE PREDICTION COMPARISON WITH VARIOUS METRICS

Task	Metric	MDN	MLP	LSTM
Pick	CC	**0.722**	0.764	0.0708
	NSS	**11.6**	11.4	0.811
	SIM	**0.629**	0.609	0.0693
	KL	**0.981**	1.15	22.5
Pick-2	CC	**0.616**	0.558	0.0242
	NSS	**7.36**	6.17	0.115
	SIM	**0.495**	0.445	0.0326
	KL	**3.13**	3.71	24.4
Pick-and-place	CC	**0.588**	0.478	0.0386
	NSS	**5.45**	4.01	0.300
	SIM	**0.473**	0.386	0.0410
	KL	**3.28**	4.15	24.9
Pick-and-place×2	CC	**0.666**	0.512	0.0853
	NSS	**5.19**	3.67	0.526
	SIM	**0.530**	0.419	0.0726
	KL	**2.08**	3.60	22.3
Pick-and-place×3	CC	**0.664**	0.585	0.0711
	NSS	**4.19**	3.39	0.335
	SIM	**0.554**	0.493	0.0666
	KL	**1.49**	1.70	22.7

D. Gaze prediction evaluation with various metrics

A comparison of the performance of the models for gaze prediction in terms of the CC, NSS, SIM, and KL on all tasks are shown in Table VII. The MDN outperformed Regression-MLP and Regression-LSTM on all tasks and all evaluation metrics. Lower values are better in KL.

REFERENCES

[1] D. A. Pomerleau, "Alvinn: An autonomous land vehicle in a neural network," in *Proc. Advances Neural Inf. Process. Syst.*, 1989, pp. 305–313.

[2] A. Billard, S. Calinon, R. Dillmann, and S. Schaal, "Robot programming by demonstration," *Springer Handbook of Robotics*. Berlin, Germany: Springer, 2008, pp. 1371–1394.

[3] L. Rozo, P. Jiménez, and C. Torras, "A robot learning from demonstration framework to perform force-based manipulation tasks," *Intell. Service Robot.*, vol. 6, no. 1, pp. 33–51, 2013.

[4] C. Lynch *et al.*, "Learning latent plans from play," *Conf. Robot Learn.*, 2019. [Online]. Available: https://learning-from-play.github.io

[5] S. Gu, E. Holly, T. Lillicrap, and S. Levine, "Deep reinforcement learning for robotic manipulation with asynchronous off-policy updates," in *Proc. Int. Conf. Robot. Autom.*, 2017, pp. 3389–3396.

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

[7] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. Advances Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[8] R. Girshick, "Fast r-cnn," in *Proc. Int. Conf. Comput. Vision*, 2015, pp. 1440–1448.

[9] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2016, pp. 770–778.

[10] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2016, pp. 779–788.

[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] 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. Autom.*, 2016, pp. 512–519.

[13] R. Rahmatizadeh, P. Abolghasemi, L. Bölöni, and S. Levine, "Vision-based multi-task manipulation for inexpensive robots using end-to-end learning from demonstration," in *Proc. Int. Conf. Robot. Autom.*, 2018, pp. 3758–3765.

[14] A. Nair, D. Chen, P. Agrawal, P. Isola, P. Abbeel, J. Malik, and S. Levine, "Combining self-supervised learning and imitation for vision-based rope manipulation," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2017, pp. 2146–2153.

[15] S. Levine, P. Pastor, A. Krizhevsky, J. Ibarz, and D. Quillen, "Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection," *Int. J. Robot. Res.*, vol. 37, no. 4-5, pp. 421–436, 2018.

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

[17] J. Pelz, M. Hayhoe, and R. Loeber, "The coordination of eye, head, and hand movements in a natural task," *Experimental Brain Res.*, vol. 139, pp. 266–77, 2001.

[18] C. M. Bishop, "Mixture density networks," Neural Computing Research Group, Aston Univ., Birmingham, U.K., Tech. Rep. NCRG/94/004, 1994.

[19] P. Abbeel and A. Y. Ng, "Apprenticeship learning via inverse reinforcement learning," in *Proc. Int. Conf. Mach. Learn.*, 2004, pp. 1–8.

[20] B. D. Ziebart, A. Maas, J. A. Bagnell, and A. K. Dey, "Maximum entropy inverse reinforcement learning," *Assoc. Advancement Artif. Intell.*, vol. 8, pp. 1433–1438, 2008.

[21] P. Abbeel, A. Coates, and A. Y. Ng, "Autonomous helicopter aerobatics through apprenticeship learning," *Int. J. Robot. Res.*, vol. 29, no. 13, pp. 1608–1639, 2010.

[22] S. Levine, Z. Popovic, and V. Koltun, "Nonlinear inverse reinforcement learning with gaussian processes," in *Proc. Advances Neural Inf. Process. Syst.*, 2011, pp. 19–27.

[23] I. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. Advances Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.

[24] J. Ho and S. Ermon, "Generative adversarial imitation learning," in *Proc. Advances Neural Inf. Process. Syst.*, 2016, pp. 4565–4573.

[25] X. B. Peng, A. Kanazawa, S. Toyer, P. Abbeel, and S. Levine, "Variational discriminator bottleneck: Improving imitation learning, inverse rl, and gans by constraining information flow," in *Proc. Int. Conf. Learn. Representations*, 2019. [Online]. Available: https://xbpeng.github.io/projects/VDB/2019_VDB.pdf

[26] 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.

[27] M. Laskey, J. Lee, R. Fox, A. Dragan, and K. Goldberg, "Dart: Noise injection for robust imitation learning," in *Proc. Conf. Robot Learn.*, 2017, pp. 143–156.

[28] N. Liu, J. Han, D. Zhang, S. Wen, and T. Liu, "Predicting eye fixations using convolutional neural networks," in *Proc. Conf. Comput. Vision Pattern Recognit.*, 2015, pp. 362–370.

[29] J. Pan, E. Sayrol, X. Giró-i Nieto, K. McGuiness, and N. E. O'Connor, "Shallow and deep convolutional networks for saliency prediction," in *Proc. Conf. Comput. Vision and Pattern Recognit.*, 2016, pp. 598–606.

[30] S. Jetley, N. Murray, and E. Vig, "End-to-end saliency mapping via probability distribution prediction," in *Proc. Conf. Comput. Vision Pattern Recognit.*, 2016, pp. 5753–5761.

[31] M. Kümmerer, L. Theis, and M. Bethge, "Deep gaze i: Boosting saliency prediction with feature maps trained on imagenet," in *Proc. Int. Conf. Learn. Representations*, 2015, pp. 1–12.

[32] X. Huang, C. Shen, X. Boix, and Q. Zhao, "Salicon: Reducing the semantic gap in saliency prediction by adapting deep neural networks," in *Proc. Int. Conf. Comput. Vision*, 2015, pp. 262–270.

[33] L. Bazzani, H. Larochelle, and L. Torresani, "Recurrent mixture density network for spatiotemporal visual attention," *Int. Conf. Learn. Representations*, 2017. [Online]. Available: https://openreview.net/pdf?id=SJRpRfKxx

[34] H. O. Latif, N. Sherkat, and A. Lotfi, "Fusion of automation and teleoperation for person-following with mobile robots," in *Proc. Int. Conf. Inf. Autom.*, 2009, pp. 1240–1245.

[35] H. Admoni and S. Srinivasa, "Predicting user intent through eye gaze for shared autonomy," in *Proc. Assoc. Advancement Artif. Intell. Fall Symp. Ser.*, 2016, pp. 298–303.

[36] C. Carreto, D. Gêgo, and L. Figueiredo, "An eye-gaze tracking system for teleoperation of a mobile robot," *J. Inf. Syst. Eng. Manage.*, vol. 3, no. 2, pp. 16–24, 2018.

[37] D. H. Yoo, J. H. Kim, D. H. Kim, and M. J. Chung, "A human-robot interface using vision-based eye gaze estimation system," in *Proc. IEEE Int. Conf. Intell. Robots Syst.*, 2002, vol. 2, pp. 1196–1201.

[38] O. Palinko, F. Rea, G. Sandini, and A. Sciutti, "Robot reading human gaze: Why eye tracking is better than head tracking for human-robot collaboration," in *Proc. IEEE Int. Conf. Intell. Robots Syst.*, 2016, pp. 5048–5054.

[39] R. Zhang *et al.*, "Agil: Learning attention from human for visuomotor tasks," in *Eur. Conf. Comput. Vision*, 2018, pp. 663–679.

[40] M. Kummerer, T. S. Wallis, and M. Bethge, "Saliency benchmarking made easy: Separating models, maps and metrics," in *Proc. Eur. Conf. Comput. Vision*, 2018, pp. 770–787.

[41] Z. Bylinskii, T. Judd, A. Oliva, A. Torralba, and F. Durand, "What do different evaluation metrics tell us about saliency models?" *Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 3, pp. 740–757, 2018.

[42] T. Judd, F. Durand, and A. Torralba, "A benchmark of computational models of saliency to predict human fixations," MIT Tech. Report, Computer Science and Artificial Intelligence Laboratory, Cambridge, MA, USA, Tech. Rep. MIT-CSAIL-TR-2012-001, 2012.

[43] R. J. Peters, A. Iyer, L. Itti, and C. Koch, "Components of bottom-up gaze allocation in natural images," *Vision Res.*, vol. 45, no. 18, pp. 2397–2416, 2005.

[44] L. Itti and P. F. Baldi, "Bayesian surprise attracts human attention," in *Proc. Advances Neural Inf. Process. Syst.*, 2006, pp. 547–554.

[45] L. Liu *et al.*, "On the variance of the adaptive learning rate and beyond," *Int. Conf. Learn. Representations*, 2020. [Online]. Available: https://openreview.net/forum?id=rkgz2aEKDr