2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

Multi-Fingered Dragging of Unknown Objects and Orientations Using Distributed Tactile Information Through Vision-Transformer and LSTM

T. Ueno, S. Funabashi, H. Ito, A. Schmitz, S. Kulkarni, T. Ogata, and S. Sugano

Abstract— **Multi-fingered hands can be suitable for stable object manipulation. Furthermore, abundant tactile information can be acquired with multi-fingered hands, useful to recognize the object's properties, which is beneficial to adapt the motion to the object. However, generating dexterous manipulation motions with multi-fingered hands with high density tactile sensors is challenging due to complex touch states. Hence, tasks that conventionally require a high level of active tactile sensing simultaneously with motion generation, such as pulling in the hand while recognizing the posture of an object are difficult to accomplish. In this letter, we propose a novel deep predictive learning approach using Vision-Transformer (ViT) and Long-Short Term Memory (LSTM). The ViT's attention mechanism can spatially focus on specific fingers represented by distributed 3-axis tactile sensors (uSkin). The LSTM can preserve long time-series information of the manipulation which can realize changing the desired motion according to the initial touching position and orientation for the target object. Results showed that the ViT-LSTM is effective in performing adaptive finger movements according to the properties of the object, i.e. its hardness and relative posture.**

I. INTRODUCTION

Humans do not necessarily reach an object accurately when grasping it, but often use their fingers in synchrony to change the position and the orientation of the object to compensate for the positional difference. Multi-fingered hands enable such error compensation dexterously [1]. Humans rely on touch states of objects and achieve dexterous tasks by making finger movements adaptive to such touch states. Hence tactile sensing plays an important role in recognizing the object states and touch states [2]. However, generating dexterous multiple-finger motions using tactile sensors with dynamically changing touch states is challenging . In our previous works, processing tactile information spatially was verified to be beneficial for successfully achieving difficult multi-fingered tasks (e.g. object recognition [3] and in-hand manipulation [4]). Tactile data from sensors attached to the fingers and palms of an Allegro Hand was processed by a graph convolutional network (GCN), where the array of distributed taxels were represented as graph. However, there were some limitations in the generalization ability of the model. The initial grasping position of the target objects had to be fixed, and an object property label was required for successful manipulation with daily objects [4].

This research was supported by the Japan Science and Technology Agency, ACT-I Information and Future Acceleration Phase with a grant number of JPMJPR18UP and Moonshot R&D with a grant number of JPMJMS2031.

The authors are with Waseda University, Okubo 3-4-1, Shinjuku, Tokyo 169-8555, Japan. (e-mail: s-funabashi@aoni.waseda.jp).

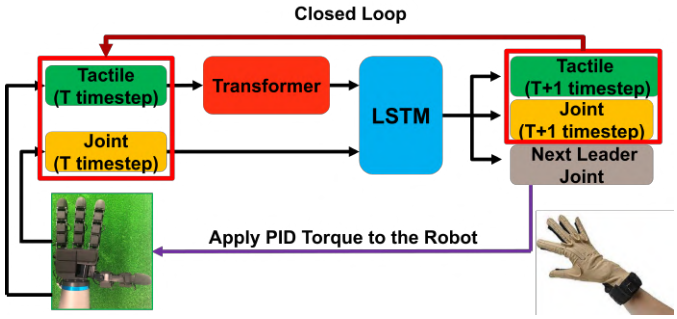

Fig. 1. Schematic of the proposed motion-generating method.

In the case of a multi-fingered hand, multiple fingers need to move concurrently, according to diversely different touch states to produce a coherent movement to achieve manipulation. For such a scenario it might be beneficial to predict which part of the hand to focus on, i.e. the input sensory data at a given timestep. Furthermore, since the need to specify object property labels implies limitations on being adaptive to the diversity of the objects, it would be beneficial to recognize and extract the properties and states of the objects without having to explicitly input them. One study demonstrated that two robot arms achieved dexterous manipulation by an attention mechanism which focused on either arm at each manipulation phase [5]. Being able to focus on different parts of hand by learning attention might be beneficial in recognizing the object properties and states relevant to the manipulation task. Therefore, we propose that an attention mechanism for the tactile data can potentially enhance the performance of dexterous tasks with a multi-fingered hand. Moreover, a certain finger motion for each object state and property should be autonomously chosen and generated to achieve grasping objects. Deep predictive learning has been used for adaptive motion generation to object properties by remembering sensor information. A multi-fingered hand can also achieve adaptive motions by remembering touch states and properties in a deep predictive learning manner.

Hence, our work proposes utilizing (1) Vision Transformer (ViT) [6], which has recently been shown to be effective in the fields of image processing, for providing a spatial attention mechanism to determine which part of hand to focus on during the manipulation. Furthermore, we use (2) long-short term memory (LSTM) [7], which is widely used for motion generation in the robotics field [8]. It can keep long time-series information so that after an initial

exploratory movement of touching the object, a finger motion can be generated adaptive to the unknown object's orientation and properties. The input of Transformer-LSTM model is the tactile data and joint angles in the manipulation task.

To evaluate the proposed method, we define the task of dexterously dragging objects placed away from the hand according to their orientations and properties. This task consists of the following two phases: (1) recognizing the orientation and properties of the object by touching the object and acquiring tactile data, and dexterously rotating it with the fingers to correct its orientation. This cannot be achieved by conventional methods [4] or methods where object properties need to be explicitly specified. (2) Dragging the object up to the palm and grasping it based on tactile information with the fingers dexterously, which requires coordinating the fingers and making dexterous movements. Finally, this paper presents following contributions:

- A deep predictive learning approach using ViT and LSTM for multi-fingered in-hand manipulation.
- Evaluating the effectiveness of the proposed ViT-LSTM through an ablation study.
- Analysis of the internal representations of the proposed ViT-LSTM showing the effectiveness of ViT and LSTM.
- Evaluating the effectiveness of the ViT-LSTM comparing it to open-loop motion generation without using machine learning for unknown objects.

II. Related Work

A. Control Method for Multi-Fingered Manipulation

Many multi-fingered manipulation strategies using fingertips with kinematics and dynamics were proposed before 2000 [9]. This way usually requires information about the grasped objects and the environment in advance to achieve dexterous manipulation [10]. However, mathematical modeling of manipulation with target objects and an environment precisely is difficult when both objects and environment need to be described together, resulting in unpredictable scenarios. Therefore, model-based methods had some drawbacks in the real world. On the other hand, learning-based methods can acquire the policies which can be adaptive for such complicated manipulation situations from training data without prior modeling of the external environment. Reinforcement learning is powerful for dynamic manipulation tasks [11]. However, generating precise motions is still difficult and it is not adaptive to diverse objects such as soft objects [12]. One study achieved a manipulation task even without tactile sensors with a variety of objects [13], but the study was limited to similar objects (sizes and shapes) and used only fingertips. Imitation learning is also effective to build a policy to execute multi-fingered manipulation with a data collection method via human tele-operation [14]. [15] used imitation learning for a variety of multi-fingered manipulations, but the tasks were done in simulation and not adaptive to object properties.

B. Tactile Sensing and Spatial Processing

Tactile sensors have supported control methods to achieve tasks stably [16]. Many previous studies used spatial processing methods such as convolutional neural networks (CNNs) for tactile information effectively [17], while not many studies have successfully incorporated tactile sensing skills to achieve multi-fingered manipulation tasks. [4] dealt with picking with dynamic finger motions and achieved the tasks with various real objects using GCNs. Although the method got tactile features geodesically following the robot hand shape and achieved dexterous manipulation with a high success rate, it was not robust to initial grasping states (i.e. positions and orientations of objects) as fingers motion could drastically change depending on the states. A study [18] has revealed that recognizing object positions and orientations were useful for tele-manipulation. Being adaptive to the object states can still be important and is an unresolved issue for robotic manipulation. On the other hand, Transformer [19], which has an attention mechanism, was applied to dual-arm manipulation where collaborative motions with both arm was necessary [5]. This idea is a key for multi-fingered hands when we consider them as multiple robot arms. Furthermore, Vision Transformer (ViT) is the mainstream in the field of image processing [6]. [20] used ViT to handle image and tactile information for slip detection of objects. [21] succeeded in manipulating rigid objects by extracting the features of image and touch sensor information using ViT. For the case of multi-fingered tasks, high-dimensional distributed tactile sensors are suitable for manipulating soft objects that can detect the entire deformation of the objects. Therefore, ViT has potential for processing such tactile information spatially.

C. Active Sensing for Time-Series Information

Although skillful manipulation with tactile sensing was achieved, an adaptive motion generation to a variety of situations requires recognizing not only the type of object but also the grasping states. Recurrent neural networks were also used to achieve a grasping task [22]. In-hand manipulation with tactile sensors on fingertips was achieved [23] and [24] used a CNN-LSTM. Although they could achieve dexterous manipulation tasks, manipulation motions were not actively changed depending on object properties or orientations. On the other hand, deep predictive learning is one of the possible methods to achieve such adaptive manipulation. This method estimates near-future sensor values from the current sensor values recursively [8]. In the case of conventional deep learning methods, it is difficult to remember the object states and the corresponding motions. However, in the case of deep predictive learning, the robot can generate long manipulation motion consisting of several sub-manipulation motions. Also, tactile information can be actively sensed to remember deformability of objects in a time series [25]. Thereby, adaptive multi-fingered manipulation based on object states represented by tactile information can be achieved by active tactile sensing.

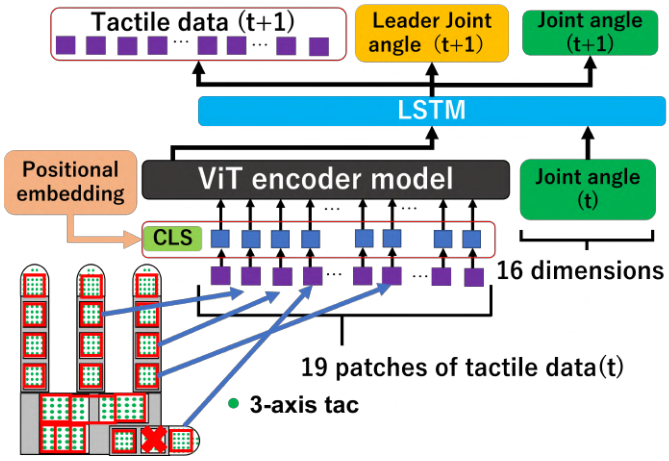

Fig. 2. Detailed proposed ViT-LSTM architecture.

III. Proposed Method

A. Allegro Hand with Tactile Sensors

An Allegro Hand, a multi-fingered robotic hand from Wonik Robotics, which provides 16 DOFs (each finger has 4 DOFs) was used. The uSkin distributed tactile sensors incorporated into the Allegro Hand was made by XELA Robotics Inc. The uSkin provides 3-axis tactile measurements with small magnets located above sensor chips embedded in soft material. The uSkin is mounted on the fingertips, finger phalanges and palm as shown in Fig. 2. Note that one sensor patch was broken (cross mark in Fig. 2) during our experiments and was not used for the evaluation experiments in Section V. Regarding this point, ViT is adaptive to a change of number of sensor patches (detailed explanation in Section III-B) and the position of the patch was where an object rarely touched in the experiments. Therefore, the experiments were conducted without any critical problems. Overall, the following measurements were provided from the robot hand hardware: 16 (DoF) + 912 (19 patches * 16 sensor chips * 3axes : extracted from 1056 sensor chips measurements) = 928 measurements. This sensor information is collected at a speed of 20 Hz. Some tactile measurements were not used for inputting to the ViT-LSTM (explained in Section III-B).

B. Data Collection

In case of imitation learning, teaching a robot any motion requires a human to create the said motion data and train a model with it. The Allegro Hand was controlled by transferring the joint angles of a dataglove (CyberGlove 3) to the AllegroHand. In the control of the Allegro Hand, joint torque values required to achieve the target joint position were solved by PID control.

C. ViT: Attention Mechanism and Distributed Tactile Sensors

Fig. 2 shows the entire structure of the proposed model. Firstly, the ViT encoder consists of the following elements; (i) embedded patches, (ii) layer normalization, (iii) residual connection, (iv) multi-head attention and (v) MLP (multi-layer perceptron). The values of the 19 tactile sensor patches of the Allegro Hand are firstly input to the ViT encoder. In the embedded patches phase, the process of dividing the tactile data into patches before inputting them to the ViT encoder model is conducted at a linear transformation layer and a positional embedding layer. The input patch from the sensor patch on the phalanges on the Allegro Hand has a shape of 4 (height) x 4 (width) x 3 (x, y and z) input. In order to make same-sized Transformer's tokens, we cropped 4×4 sensor patches shown in the red box on the Allegro Hand of Fig. 2. Note that since the patch structure of the Allegro Hand cannot be neatly divided into 4 x 4 patches, one of the palm patches is overlaid as inputs for two tokens so as not to reduce the sensor information. Each sensor patch is allocated to the input patches with linear transformation. Unlike a usual ViT, we did not share weights in the linear transformation of each sensor patch due to too much variability of the uncalibrated tactile sensor measurements, which the ViT-LSTM seems not to be able to adapt to. Next, image classification tokens called CLS tokens are added to the input patches, and then they are flattened. The positional information of the tokens which is a vector defined by trigonometric functions is added to the flattened input. Finally, the shape of the input data z_0 is

$$\mathbf{z_0} = [\mathbf{x}_{class}; \mathbf{x}_p^1\mathbf{E}_p^1; \mathbf{x}_p^2\mathbf{E}_p^2; \cdots; \mathbf{x}_p^N\mathbf{E}_p^N] + \mathbf{E}_{pos} \quad (1)$$

where x_{class} is the learnable CLS token, x_p^n is a n_{th} vector for a patch and E_{pos} is sinusoidal positional encoding layer. The z_0 is passed to the encoder of the Transformer encoder. In the Transformer encoder, layer normalization is used as a method that can robustly suppress gradient loss even when the data distribution between mini-batches deviates. The residual connections are also used to solve the problem of vanishing gradients in very deep network architectures. The multi-head attention generates an attention between the tokens, and transforms the token representation based on the relationship between the tokens. When applying an attention between tokens, a scaled dot product which is expressed by the following equation is used.

$$Z_i = softmax(\frac{Q_i K_i^T}{\sqrt{d_k}})V_i \quad (2)$$

where the inner product (similarity) of Q_i (a query matrix) and K_i (a key matrix) is calculated and V_i (a value matrix) is multiplied with the corresponding weights. $\sqrt{d_k}$ is a scaling factor that prevents the inner product from becoming too large.

By processing the above calculation multiple times, the attention Z_i is obtained as multiple heads. They are combined as in the following equation,

$$MultiHeadAttention(Q,K,V) = concat(Z_1, Z_2, .., Z_N)W^O \quad (3)$$

where W^O is a weight matrix for the output of the multi-head attention. Although multi-head attention was originally used for ViT, we heuristically employed single-head attention in our ViT-LSTM which outperformed multi-head attention in terms of the success rate of in-hand manipulation.

The MLP layer receives the output token from the multi-head attention.

D. Deep Predictive Learning and Imitation Learning

Next, the output of the CLS tokens of the ViT encoder which is the learned representation of the entire tactile information is input to the LSTM layer together with the joint angles. Then, the LSTM outputs the next time step of the tactile information, the joint angles of the Allegro Hand and the leader joints which are a remapped target joint angles from the dataglove to the Allegro Hand. The joint angle and tactile data predicted from ViT-LSTM are again used as input in the next timestep with a weighted average between the raw data (joint angle and tactile data from Allegro Hand). The higher the ratio of this weighted average (called closed loop ratio), the more stable the prediction, whereas a smaller closed loop ratio can make the model more adaptive [8]. Subsequently, they are input to the ViT-LSTM to generate in-hand manipulation. For this specific task, we empirically determined the closed loop ratio to be 0.5. Unlike [4], this study uses the leader joint angle output of the ViT-LSTM to control the Allegro hand. In this case, the human motions are directly imitated to manipulate the target objects instead of using the observed joint angles from the robot hand. This is because the produced torques from the observed joint angles of the robot hand were usually too weak to successfully drag and grab objects. In particular, during collecting of training data, the robot hand produces sufficient torques with our mapping of the leader joint angles to the robot hand. However, the applied forces to the object will be lost when only looking at the joint angles of the robot hand afterwards.

E. Loss Function

The loss function consists of sum of the Mean Squared Error (MSE) losses corresponding to three modalities of outputs predicted by the model, joint (j), desired joint (dj), and tactile (tc). A modality coefficient k_m is applied to the loss before summing up, where m corresponds to the respective modality, $m \in \{j, dj, tc\}$. Through experimentation, we concluded the value of k_m to be 0.1. Hence, the loss function for total loss, E is defined as:

$$E_{total} = \sum_{m \in \{j,dj,tc\}} \sum_{t \in T} \{k_m \times E_m(t)\} \tag{4}$$

$$E_m(t) = \frac{1}{N_m} \parallel y_m(t) - d_m(t+1) \parallel^2 \tag{5}$$

T denotes the total number of timesteps for to an action, while t represents the present timestep. E_m denotes loss corresponding to a modality and E_{total} corresponds to the total loss. The term d_m corresponds to the datum associated with the respective modality. N_m corresponds to input size of the respective modality, which is 16 in case of joint and desired joint, 912 in case of tactile as there are 912 tactile measurements in the input data.

IV. Experiment Design

A. Collecting Data

The manipulation strategy can vary from object to object, and we planned the strategy for the data collection similar to how humans would operate. For example, for a long object,

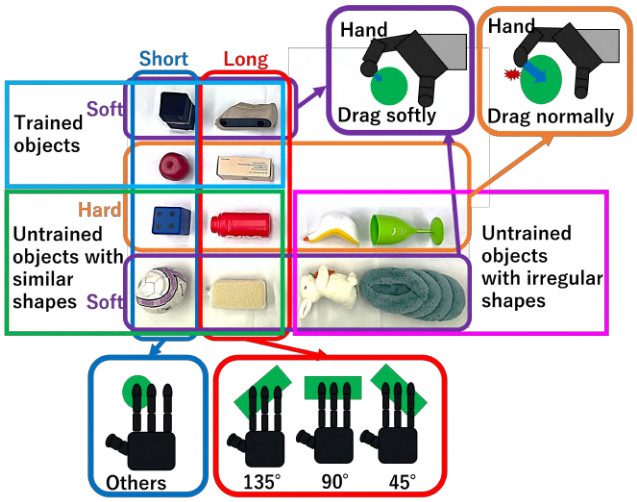

Fig. 3. Target objects and object orientations and properties. Trained objects: A box made of corrugated paper was chosen as a long and hard object. A petbottle bag was chosen as a long and soft object. A stuffed guy apple was chosen as an short and hard object. A sponge was chosen as a short and soft object. Since the box and petbottle bag are long, they have 3 different initial grasping orientations (135, 90 and 45 degrees). Untrained objects: A bottle can was chosen as a long and hard object. Another sponge was chosen as a long and soft object. A stuffed dice was chosen as an short and hard object. A large ball was chosen as a short and soft object. The bottle and sponge are long and they also have 3 different initial grasping orientations (135, 90 and 45 degrees)

humans will likely tend to correct their orientation before dragging, but for shorter objects, orientation would matter only a little. For soft objects, the dragging force will be deliberately small as opposed to hard objects.

Before the dragging motion, in order to obtain the posture of the object and its internal properties, the hand touches the object by lowering its fingers. After that, the hand moves up from the object, and then the allocated motion in accordance with the properties of the object is initiated. The model is trained to perform a series of actions, from the initial motion towards touching the object to the grasping motion with the target object. As the target motion in this study, the target object with a certain orientation (shown in Fig. 3) is dragged through a hooking motion by the fingers and then finally grasped into the hand.

When the initial posture of the object is 135 degrees, the object is rotated by repeatedly pulling the object with the little finger so that the object is aligned with the palm. Afterwards, the fingers drag the object closer to the palm and then the object is lifted. When the object initial orientation is 90 degrees, the object is repeatedly pulled with the middle finger to achieve aligning to the palm. When the initial orientation is 45 degrees, the index finger is used for repeating pulling motions to change the posture. For other objects, the stuffed apple which is spherical and easily pushed away by fingers is pulled into the palm by slowly dragging the object with the middle and index fingers. The sponge as a cube shaped object is rotated by moving the middle and index fingers alternately. In all cases, the object is firmly grasped by all fingers at the end of the entire motion.

TABLE I

NEURAL NETWORK SETTING

Input	Follower Joint	16	
	Tactile	19 (Patches) * 48	
ViT block	Num of Neurons (ViT)	20	
	Num of Heads	1	
	MLP Layer	Num of Neurons	64
		Activation Function	GELU
	Num of Encoder	3	
LSTM block/ FC Layer	Input Size	20(ViT Features) + 16 (Joint)	
	Output Size	500	
Output	Follower Joint	Output Size	16
		Activation Function	LeakyReLU
	Leader Joint	Output Size	16
		Activation Function	LeakyReLU
	Tactile	Output Size	912
		Activation Function	LeakyReLU

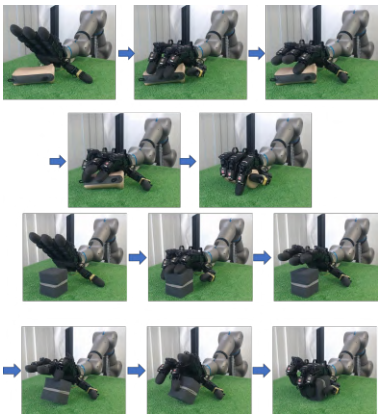

Fig. 4. Example of successful in-hand manipulation with a petbottle bag at 135 degrees (top) and a sponge (bottom).

The soft objects cannot be dragged into the hand easily because when trying to drag the object with strong force, the object will be pressed against the artificial lawn and does not move due to a large friction against the lawn. Therefore, the dataset for the petbottle bag was collected with weak force.

In this experiment, we prepared 60 trials per object, resulting in a total of 240 trials for four objects. For long objects, we prepared 20 trials for each initial pose, and for short objects, we prepared 60 trials, always with the same pose. In addition, 192 of the 240 trials used for training dataset, while the remaining 48 trials were used as validation dataset. Tactile and joint information were recorded at a rate of 20Hz. Each trial was recorded for 31.5 seconds, resulting in 630 timesteps for each sequence.

B. Neural Network Settings

To evaluate the effectiveness of our proposed ViT-LSTM model, we compared with two other models: an LSTM-only model and a ViT-only model. We investigated the success rates of these models for the dragging task performed on trained objects. The parameter setting of the models is described in Table I. The input to all the three models consisted of tactile information with 912 dimensions and joint angle with 16 dimensions. The models were designed to output tactile information for the next timestep with 912 dimensions, as well as joint angle information and leader joint angle information with 16 dimensions, respectively. To ensure the sensitivity to the input information, joint angle and tactile information were normalized between 0.1 and 0.9 for each joint and tactile values, respectively. Our proposed ViT-LSTM model is composed of a ViT-encoder, followed by an LSTM, and an fully-connected layer. The tactile information is first processed by the ViT. The total size of the input is 19 tokens × 48 dimensions, and each token is independently converted to a 20-dimensional embedding. The output is then processed by a 3-layered ViT encoder to extract the output of the CLS token. The attention module in each ViT encoder uses single-head attention. The MLP employs the GELU activation function, and the hidden layer size is 64 dimensions. The 20-dimensional feature extracted by the CLS token is added to the 16-dimensional joint angle information to obtain a 36-dimensional feature vector, which is then fed into a 500-dimensional hidden layer LSTM. The output of the LSTM is then used to predict the 912-dimensional tactile information, as well as the 16-dimensional joint information and 16-dimensional leader joint information. LeakyReLU was used as activation function. During training, the model was trained using the AdamW optimizer with a learning rate of 0.0001.

In the case of the LSTM-only model, we replaced the ViT block of the proposed ViT-LSTM with an FC Layer to extract 20-dimensional features from the tactile data of 912 dimensions in a single layer. Similarly, in the case of the ViT-only model without LSTM, we replaced the LSTM block of the proposed ViT-LSTM model with an FC Layer to expand the 36-dimensional feature vector to 500 dimensions.

V. EVALUATION

A. Ablation Study with Proposed Model

In order to demonstrate the effectiveness of the proposed ViT-LSTM, 5 trials were conducted for each object and the initial grasping orientations. The success conditions for the dragging motion with an object placed at a distance from the hand are defined as follows:

1) The posture of the long object after grasping is within $\pm$15 degrees relative to the horizontal direction of the palm.
2) The object does not fall after the table moves away from the hand.

Table II shows the result of success rates for the target in-hand manipulation with the 3 models. The ViT-LSTM could achieve the highest success rates. The ViT only model could not achieve the manipulation with any object. The LSTM only model could achieve the manipulation in 30 % of all the trials. From this result, time-series information was useful as initial contacts to an object was a key to decide a dragging motion and had to be remembered to achieve the entire manipulation. Besides, the attention mechanism in spatial manner was also important to recognize touch states from distributed tactile sensors. Finally, we could confirm that temporal information and spatial attention mechanism

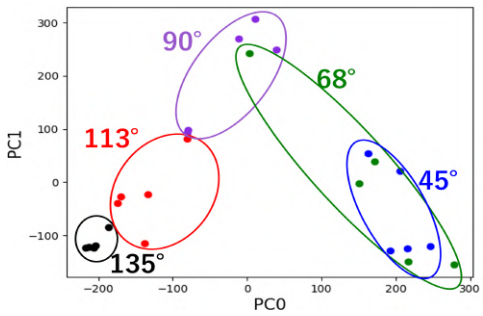

Fig. 5. A PCA map of features from the LSTM block for object orientations. All timesteps for each manipulation trial was plotted as one dot in the map. Each color represents one orientation, respectively. The untrained orientations (68 and 113 degrees) are mapped between trained orientations, and thus it was confirmed that the ViT-LSTM could recognize the orientations of the objects.

for tactile information were effective for successful in-hand manipulation.

B. Adaptability to initial grasping orientations

In this experiment, the adaptability to the object orientations was evaluated. The training dataset included 45, 90 and 135 degrees of the object orientations. We prepared 68 and 113 degrees as test orientations which were within the orientations of the training dataset and not exactly the middle orientation so that the test orientations became more difficult. Table III shows the experimental result. The proposed method was verified to be robust to achieve manipulation with high success rates at any orientation. Moreover, the features generated from the output of the LSTM blcok of the proposed method is visualized by principal component analysis (PCA) shown in Fig. 5. From this result, the ViT-LSTM acquired the features of the orientations and achieved successful in-hand manipulation even with untrained orientations.

C. Analysis on Attention Score

Fig. 6 shows the changes in attention scores of the tactile information on each finger of the Allegro Hand during the manipulation. The graphs were created by subtracting the initial attention score from all recorded attention scores, allowing a clearer understanding of the variations in attention scores over time. When the object was placed at a 135 degrees, the attention score of the little finger exhibited a sudden increase upon contact with the object. Similarly, when the object was positioned at 45 degrees, the attention score of the index finger showed a notable rise, while the middle finger's attention score exhibited a significant jump when the object was located at 90 degrees. These findings reveal that the attention mechanism enhances the score of the contact position on each finger. However, the middle finger was often observed to have a high attention score for several time-steps, and the reason for this has to be further investigated in future work.

For the internal properties of the objects, comparing the graphs of the box and a petbottle bag, we found that the magnitude of the palm's attention score after dragging the

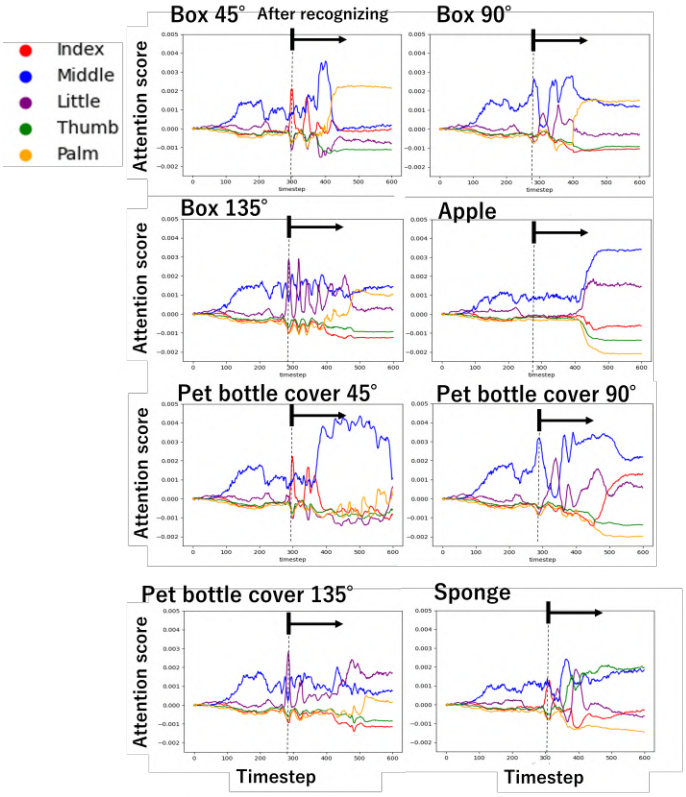

Fig. 6. Trajectories of attention scores for tactile information from each finger for one single manipulation trial.

object was larger when grasping the hard object. This can be due to the object softness. Comparing the graphs for the apple and the sponge, we found that the attention score changed more slowly when the apple was grasped. This seems to be because the Allegro Hand slowly dragged the apple with its index and middle fingers so that the apple was not pushed away by the hand.

Overall, the attention scores seem to follow changes in tactile information and manipulation motions and they could achieve successful in-hand manipulation.

D. Analysis on Features of LSTM

As shown in Fig. 7, the object orientations, in other words, the initial touch states of the fingers corresponding to each in-hand manipulation motion for each target object are shown as each cluster. From the feature map made of PC0 and PC1, the orientation of the objects corresponding to the used finger seems to be extracted for the LSTM block.

On the other hand, the map of PC0 and PC2 shows a different tendency especially that the PC2 seems to extract features of the objects. Each colored cluster becomes clearer in the PC2 axis, while the relationship of the corresponding finger for each orientation becomes vague in the PC0 axis. For example, the cluster of petbottle bag (light blue dots) and box (red dots) does not overlap each other. However since the orientation is the same (90 degrees), the clusters are close to each other. From this result, the ViT-LSTM seems to acquire the features not only for object orientations but

TABLE II

IN-HAND MANIPULATION RESULT WITH TRAINED OBJECTS

	Box 45°	Box 90°	Box 135°	Petbottle Bag 45°	Petbottle Bag 90°	Petbottle Bag 135°	Apple	Sponge	Success Rate
ViT-LSTM	5/5	5/5	5/5	5/5	3/5	5/5	5/5	5/5	95%
LSTM only	2/5	0/5	5/5	1/5	0/5	1/5	3/5	0/5	30%
ViT only	0/5	0/5	0/5	0/5	0/5	0/5	0/5	0/5	0%

TABLE III

IN-HAND MANIPULATION AT DIFFERENT INITIAL GRASPING POSITIONS

	Box 45°	Box 68°	Box 90°	Box 113°	Box 135°	Success Rate
ViT-LSTM	5/5	4/5	5/5	5/5	5/5	96%

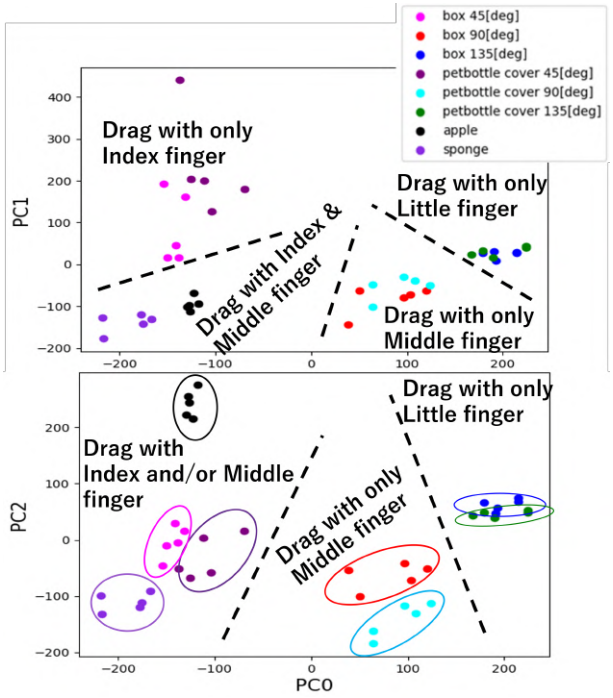

Fig. 7. A PCA map of features from the LSTM block for corresponding fingers and object states. Top figure shows a map for PC0 and PC1 where clusters for the corresponding fingers to the object orientations emerge.

also the internal property (i.e. softness). With this result, the ViT-LSTM could generate the corresponding finger motions for the target object states.

E. Comparison Experiment with Open-Loop Controller

We evaluated the effectiveness and robustness of ViT-LSTM against open-loop motions.

In case of open-loop motion, we replayed the motions in training data of the object with shape and orientation similar to the unknown object. We selected eight patterns consisting of the combination of different types and orientations of objects with similar in shape to the training objects but opposite softness properties.

As shown in Table IV, the ViT-LSTM could achieve successful in-hand manipulation for these untrained objects of known shape in 78% of all cases. At 135 degrees, the bottle slid significantly between the ground and the bottle while pulling. It resulted in a difficult posture to grasp with less friction between the object and the hand and it often fell. For the long sponge and the slipper, the object could not be pulled due to the fact that the friction with the lawn was strong at 135 degrees. This resulted into mediocre success rates for said objects at said orientations.

The success rate of open-loop trajectories is significantly lower compared to ViT-LSTM, confirming the robustness of the proposed method and effectiveness for adaptability. Experiments confirmed that the method without learning often crushes the soft object during motion generation. Furthermore, we often observed cases where the the object slipped out of the hand after grasping, because the grasp was not adaptive to the object.

TABLE IV

COMPARISON OF VIT-LSTM WITH OPEN-LOOP TRAJECTORY WITH UNTRAINED OBJECTS WITH KNOWN SHAPE

	Bottle 45°	Bottle 90°	Bottle 135°	Long Sponge 45°	Long Sponge 90°	Long Sponge 135°	Large Ball	Dice	Success Rate
ViT-LSTM	2/4	3/4	3/4	4/4	4/4	2/4	3/4	4/4	78%
OpenLoop	2/4	2/4	2/4	3/4	0/4	3/4	1/4	4/4	53%

TABLE V

IN-HAND MANIPULATION RESULT FOR VIT-LSTM WITH IRREGULAR SHAPED UNTRAINED OBJECTS

	Slipper 45°	Slipper 90°	Slipper 135°	Cup 45°	Cup 90°	Cup 135°	Stuffed Animal 45°
ViT-LSTM	2/4	2/4	2/4	4/4	4/4	3/4	2/4

Stuffed Animal 90°	Stuffed Animal 135°	Duck Figure 45°	Duck Figure 90°	Duck Figure 135°	Success Rate
3/4	4/4	3/4	3/4	4/4	75 %

F. ViT-LSTM with Untrained Complex-Shaped Object

In this experiment, we selected objects that were completely different in shape from the training objects. This set was prepared as more difficult setting which has diverse and irregular shapes and different softness. The ViT-LSTM achieved 75% success rate for dragging motions as shown in Table V. From this, we confirmed that the proposed method could generalize to untrained objects with only a small amount of training data of four types of objects. Although manipulation with the complex-shaped untrained objects was achieved with high success rates, the success rates for the slipper was low. For the slipper, the object could not be

pulled because the friction with the table surface was strong.

VI. CONCLUSIONS

In this paper, we proposed a ViT-LSTM model trained in a deep predictive learning model and achieved adaptive in-hand manipulation motions corresponding to the object states (positions, orientations and softness).

The ViT block consists of an encoder part of a Transformer model using each tactile sensor patch as a single token. The LSTM was used in a deep predictive learning manner. It enabled the ViT-LSTM to generate collaborative finger motions corresponding to the initial grasping states of the target object by remembering the tactile information from the initial active sensing motion. Furthermore, by analyzing the attention-score of each finger in the ViT block of the proposed method, we found that the attention varied significantly depending on the target object and the motion. The features from the LSTM block also showed that the relationship with finger motions and object orientations and properties was acquired by the ViT-LSTM. Overall, we confirmed that our proposed model could achieve a dexterous dragging motion with a variety of unknown object states by acquiring object features.

As future work, the range of the initial orientations and positions for the target objects should be broadened. We could not include wider initial grasping positions regarding left-right misalignment from the Allegro Hand. This is because Allegro Hands have different joint configuration to that of human hands, as they do not have adduction/abduction for all fingers.

Additionally, in this experiment, data collection was conducted by using a visual feedback of human. However, using a device with haptic feedback might enhance the data quality. Furthermore, integrating diverse modalities such as images and sounds with tactile data remains a significant challenge.

REFERENCES

[1] A. Carfì, T. Patten, Y. Kuang, A. Hammoud, M. Alameh, E. Maiettini, A. I. Weinberg, D. Faria, F. Mastrogiovanni, G. Alenyà, L. Natale, V. Perdereau, M. Vincze, and A. Billard, "Hand-object interaction: From human demonstrations to robot manipulation," *Frontiers in Robotics and AI*, vol. 8, 2021. [Online]. Available: https://www.frontiersin.org/articles/10.3389/frobt.2021.714023

[2] J. R. S and F. J. Randall, "Coding and use of tactile signals from the fingertips in object manipulation tasks," *Nature Reviews Neuroscience*, vol. 10, pp. 1041–1048, May 2009.

[3] S. Funabashi, G. Yan, F. Hongyi, A. Schmitz, L. Jamone, T. Ogata, and S. Sugano, "Tactile transfer learning and object recognition with a multifingered hand using morphology specific convolutional neural networks," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–15, 2022.

[4] S. Funabashi, T. Isobe, F. Hongyi, A. Hiramoto, A. Schmitz, S. Sugano, and T. Ogata, "Multi-fingered in-hand manipulation with various object properties using graph convolutional networks and distributed tactile sensors," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 2102–2109, 2022.

[5] H. Kim, Y. Ohmura, and Y. Kuniyoshi, "Transformer-based deep imitation learning for dual-arm robot manipulation," 2021. [Online]. Available: https://arxiv.org/abs/2108.00385

[6] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," 2020. [Online]. Available: https://arxiv.org/abs/2010.11929

[7] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, pp. 1735–80, 12 1997.

[8] N. Saito, T. Shimizu, T. Ogata, and S. Sugano, "Utilization of image/force/tactile sensor data for object-shape-oriented manipulation: Wiping objects with turning back motions and occlusion," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 968–975, 2022.

[9] C. Yu and P. Wang, "Dexterous manipulation for multi-fingered robotic hands with reinforcement learning: A review," *Frontiers in Neurorobotics*, vol. 16, 2022. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fnbot.2022.861825

[10] B. Sundaralingam and T. Hermans, "In-hand object-dynamics inference using tactile fingertips," *IEEE Transactions on Robotics*, vol. 37, no. 4, pp. 1115–1126, 2021.

[11] A. Handa, A. Allshire, V. Makoviychuk, A. Petrenko, R. Singh, J. Liu, D. Makoviichuk, K. Van Wyk, A. Zhurkevich, B. Sundaralingam, Y. Narang, J.-F. Lafleche, D. Fox, and G. State, "Dextreme: Transfer of agile in-hand manipulation from simulation to reality," 2022. [Online]. Available: https://arxiv.org/abs/2210.13702

[12] X. Lin, Y. Wang, J. Olkin, and D. Held, "Softgym: Benchmarking deep reinforcement learning for deformable object manipulation," 2021.

[13] H. Qi, A. Kumar, R. Calandra, Y. Ma, and J. Malik, "In-hand object rotation via rapid motor adaptation," 2022. [Online]. Available: https://arxiv.org/abs/2210.04887

[14] I. Guzey, B. Evans, S. Chintala, and L. Pinto, "Dexterity from touch: Self-supervised pre-training of tactile representations with robotic play," 2023.

[15] D. Jain, A. Li, S. Singhal, A. Rajeswaran, V. Kumar, and E. Todorov, "Learning deep visuomotor policies for dexterous hand manipulation," in *2019 International Conference on Robotics and Automation (ICRA)*, 2019, pp. 3636–3643.

[16] L. Röstel, L. Sievers, J. Pitz, and B. Bäuml, "Learning a state estimator for tactile in-hand manipulation," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2022, pp. 4749–4756.

[17] W. Yuan, Y. Mo, S. Wang, and E. Adelson, "Active clothing material perception using tactile sensing and deep learning," in *arXiv:1711.00574*, 2018.

[18] A. C. Miguel, S. Oleg, T. Jonathan, F. Aleksey, K. Pavel, K. Hiroyuki, and T. Dzmitry, "Deepxpalm: Tilt and position rendering using palm-worn haptic display and cnn-based tactile pattern recognition," 2022. [Online]. Available: https://arxiv.org/abs/2204.03521

[19] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," 2017. [Online]. Available: https://arxiv.org/abs/1706.03762

[20] S. Cui, J. Wei, X. Li, R. Wang, Y. Wang, and S. Wang, "Generalized visual-tactile transformer network for slip detection," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 9529–9534, 2020, 21st IFAC World Congress. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2405896320331128

[21] Y. Chen, A. Sipos, M. Van der Merwe, and N. Fazeli, "Visuo-tactile transformers for manipulation," 2022. [Online]. Available: https://arxiv.org/abs/2210.00121

[22] H. Liang, L. Cong, N. Hendrich, S. Li, F. Sun, and J. Zhang, "Multifingered grasping based on multimodal reinforcement learning," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 1174–1181, 2022.

[23] M. Lambeta, P.-W. Chou, S. Tian, B. Yang, B. Maloon, V. R. Most, D. Stroud, R. Santos, A. Byagowi, G. Kammerer, D. Jayaraman, and R. Calandra, "Digit: A novel design for a low-cost compact high-resolution tactile sensor with application to in-hand manipulation," *IEEE Robotics and Automation Letters*, vol. 5, no. 3, pp. 3838–3845, 2020.

[24] S. Funabashi, S. Ogasa, T. Isobe, T. Ogata, A. Schmitz, T. P. Tomo, and S. Sugano, "Variable in-hand manipulations for tactile-driven robot hand via cnn-lstm," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020, pp. 9472–9479.

[25] R. Ishikawa, M. Hamaya, F. Von Drigalski, K. Tanaka, and A. Hashimoto, "Learning by breaking: Food fracture anticipation for robotic food manipulation," *IEEE Access*, vol. 10, pp. 99 321–99 329, 2022.