Proceedings of the 3rd International Conference on Mechatronics and Smart Systems
DOI: 10.54254/2755-2721/147/2025.22913

Enhancement of Position Accuracy via DPB-Embedded ACT Model for Fine-Grained Manipulation

Chengxu Yuan

Central South University, Changsha, China
1149855933@qq.com

Abstract: Since the 1960s, research on robots performing fine manipulation tasks has continuously advanced with the progress of electronic and information technologies. Despite some progress, challenges remain in areas such as the spatial perception capability of sensors, cumulative errors in imitation learning, and multimodal information fusion. The ACT (Action Chunking with Transformers) model has certain advantages in these areas, but it still faces shortcomings in complex state prediction and multi-scale, multimodal data processing. To address these issues, a Dynamic Position Bias (DPB) module is introduced to optimize action sequence outputs, enhance cross-scale feature capture and spatial perception, and improve the alignment and fusion of multimodal data. Experiments conducted on the Aloha simulation platform validate the effectiveness of the improved ACT model in two typical tasks, "Cube Transfer" and "Bimanual Insertion." Compared to the original ACT model and other baseline models, significant performance improvements were observed.

Keywords: Dexterous manipulation task, Transformer, Aloha, Dynamic Position Bias

1. Introduction

The concept of robots performing dexterous manipulation tasks can be traced back to the 1960s, inspired by the high flexibility and stability of the human upper limb [1]. With the development of science and technology, dexterous manipulation tasks have expanded to encompass a range of applications, from simple grasping to complex human-robot interaction scenarios. In fields where manual operation is either impossible or highly risky (see Figure 1), such as extravehicular operations in aerospace [2], surface processing and assembly of large machine parts [3], nerve connection repair and tumor removal in surgery [4], and precise positioning in wafer lithography [5], the accuracy of robotic arm manipulation is particularly crucial.

In recent years, with the development of imitation learning and Transformer-based algorithms, robots' fine manipulation capabilities have significantly improved. However, in handling complex tasks, several challenges remain in the software algorithm domain. Professor Billard and Professor Kragic pointed out that real-time, efficient sensor and actuator data processing, as well as dynamic adjustment of information weighting algorithms at different task stages, is one of the current technical challenges [1]. Taikizawa et al. identified issues related to noise handling in imitation learning, sensitivity to hyperparameters, and the contradiction between task adaptability and sensitivity as major challenges [6]. Additionally, the integration of multi-source information, such as visual and tactile sensors, in adaptive fine manipulation systems remains an unavoidable problem [7,8].

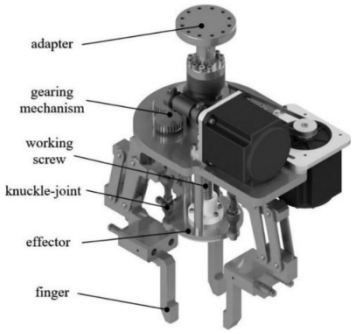
© 2025 The Authors. This is an open access article distributed under the terms of the Creative Commons Attribution License 4.0 (https://creativecommons.org/licenses/by/4.0/).

DOI: 10.54254/2755-2721/147/2025.22913

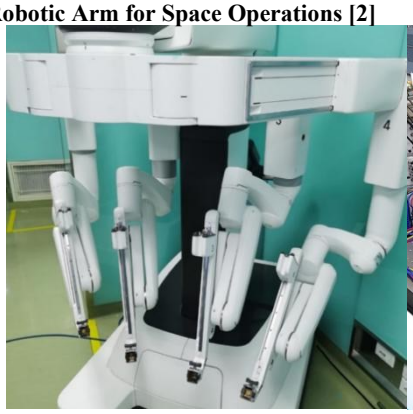
The ACT (Action Chunking with Transformers) model, proposed by Professor Zhao et al., is a data processing framework for robotic control and trajectory tracking. It processes various fine manipulation tasks that require both hands through an adaptive learning time-series generation model [9]. However, the ACT model performs poorly in complex motion prediction tasks and may still produce unpredictable responses [10]. Furthermore, the ACT model primarily focuses on fine-grained actions, tending toward single-scale motion prediction, which causes it to lack comprehensive perception of object information in complex tasks requiring multi-scale information, such as "unwrapping candy" and "Ziploc slide" [9,11]. To overcome these problems, the DPB (Dynamic Position Bias) algorithm is introduced into the ACT policy. As a key component of the CrossFormer deep learning architecture, the DPB algorithm dynamically adjusts position bias based on contextual information, enabling it to effectively capture cross-scale features in tasks and align and fuse multimodal data [12]. Moreover, compared to the ACT model, which only refers to pre-set human or scripted datasets, the DPBPL (Dynamic Position Bias Projection Layer) allows the model to adaptively adjust its understanding of position based on specific input, overcoming system errors and thereby improving position perception accuracy [12,13].



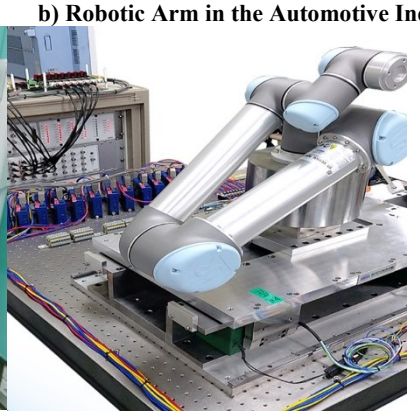
a) Robotic Arm for Space Operations [2]



b) Robotic Arm in the Automotive Industry [3]



c) Da Vinci Surgical Robot [4]



d) Wafer Handling and Transfer Robot [5]

Figure 1: Robots performing fine manipulation tasks in various fields

The key contributions of this paper are as follows:

- A novel DPB embedding strategy based on the ACT model: By incorporating dynamic position bias information into the action sequence input of the CVAE encoder, the output of the style variable z is optimized, thereby enhancing the overall fine manipulation strategy's compatibility with multi-scale information and improving positional accuracy.
- Simulation experiments based on the open-source Aloha hardware system: A series of controlled experiments were conducted using the "Cube Transfer" and "Bimanual Insertion" tasks to evaluate the performance of the optimized ACT model.

DOI: 10.54254/2755-2721/147/2025.22913

- Exploration of future research objectives: Potential directions for further enhancement of the DPB embedding strategy are proposed, with emphasis on the need to extend its application to a wider range of task types.

2. Related work

2.1. Imitation learning

Imitation learning (IL), as a method for learning skills from expert demonstrations, has shown broad application prospects in fields such as robotic dexterous manipulation and autonomous driving. Behavioural cloning is a direct supervised learning method that trains a model by observing the action sequences of an expert, aiming to make similar action choices within a certain range of random samples. Sheh et al. proposed a behaviour cloning method based on high-dimensional features to address the challenge of driving robots on rugged terrain [14]. Additionally, a method using a variational autoencoder combined with categorical latent variables was proposed. This method is applicable to tasks that involve the fusion of multiple behaviors, enabling the learning of multimodal strategies without additional environmental interactions [15]. Hsiaod et al. proposed a multi-task inverse reinforcement learning framework that clusters trajectories using the Expectation Maximization algorithm and infers the intentions of each cluster, addressing the problem of mixed behavioral demonstrations [16]. As in the strategy of integrating RGB images and joint absolute positions in the ACT model [9], multimodality learning has become one of the focal points of imitation learning in recent years. A multimodal strategy based on VAE (Variational Auto-Encoders) was proposed, which improves the learning efficiency of robots for various behaviors through classification and fusion of latent variables [15]. Furthermore, a single-step k-modal cloning method capable of effectively imitating multiple behaviors in complex environments has been studied in depth, demonstrating its potential in handling high-dimensional inputs [17].

2.2. CVAE and transformer model

CVAE (Conditional Variational Auto-Encoders), as a powerful generative model, has shown significant advantages in multimodal output distributions and flexibility. Professors Sohn and Ivanovic et al. demonstrated that CVAE can easily integrate different types of data in multi-scale predictive targets under noise injection, improving prediction accuracy while maintaining diversity [18,19]. Meanwhile, Professor Zhang designed a stacked conditional variational autoencoder (S-CVAE) architecture to capture the continuity of behavioral intentions across continuous timestamps [20]. However, these methods have certain limitations: limited multi-scale capturing ability [18], mode collapse issues [19], insufficient interpretability of the latent variable space [20], and performance bottlenecks in processing long sequences [21].

The Transformer model has shown broad application prospects in fields such as computer vision and robotics. Wang et al. addressed the limitations of traditional visual Transformers in handling multi-scale features by introducing a Cross-Scale Embedding Layer and Long Short Distance Attention [22]. In the fields of trajectory prediction and behavior learning, Researcher Zhang et al. designed a "barrier" mechanism based on the Transformer to enhance the prediction accuracy in interactive scenarios [23]; Researcher Shafiulla et al. used the Transformer decoder structure to learn diversified behaviors, significantly improving model performance [17]. However, the Transformer model still has limitations in complex tasks: effective processing of variable-size images [22], the impact of cumulative errors on overall accuracy [23], and overestimation of log-likelihood loss [17].

DOI: 10.54254/2755-2721/147/2025.22913

2.3. Dynamic position bias

Dynamic Position Bias (DPB), as an important module in the CrossFormer model, plays a role in enhancing position representation. Professor Wang et al. pointed out that compared to the traditional Relative Position Bias (RPB), which can only handle inputs of specific sizes, DPB dynamically generates position bias using a small multilayer perceptron based on the relative distance between two tokens, making it applicable to images of any size [24]. In the field of marine engineering, Værnø et al. applied DPB to position compensation for marine surface containers, thereby better capturing features at different scales, correcting bias, and improving the system's stability [13]. In the field of computer vision, Researcher Zhou et al. introduced global contextual information during the training of the DPB model and proposed the Dynamic Contextual Position Encoding algorithm, making bias handling more flexible in complex contexts and thus enhancing the adaptability and expressive power of the visual Transformer model [12].

3. Methodology

3.1. The ACT framework and mechanism

The architecture of the ACT model primarily consists of a CVAE encoder and a CVAE decoder, with the CVAE encoder being based on a transformer encoder, and the CVAE decoder composed of a set of transformer encoders and decoders.

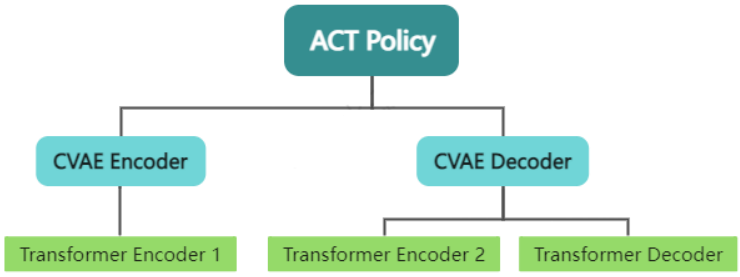


Figure 2: Diagram of the ACT model architecture

In model training, the CVAE encoder first obtains the demonstration dataset (sourced from human or scripted actions) with absolute joint positions and action sequence samples. The input data is understood through the four self-attention modules mounted on the transformer encoder, which leads to the acquisition of the style variable z. Then, four sets of RGB images, 14 absolute joint positions, and the style variable z are input into the CVAE decoder, which is understood and transformed by the transformer encoder. The transformer decoder then generates the predicted action sequence using the Action Chunking algorithm. Finally, the loss function is constructed through the mean squared error and encoder regularization loss, and the parameters of the encoder and decoder are optimized by iterating to minimize the loss function [9].

In model inference, based on the trained CVAE decoder, time-series predictions for the corresponding time step observations are generated using the RGB image and joint position information from the test dataset. To avoid uncertainty caused by the style variable during inference, it is set as the mean of a prior Gaussian distribution, i.e., the zero vector. Moreover, to enhance the smoothness and accuracy of the predicted sequence, Temporal Ensemble is used to weight "the prediction for time step t based on observations from different time steps," thus obtaining accurate action sequence predictions for time step t [9].

DOI: 10.54254/2755-2721/147/2025.22913

3.2. The embedding policy of DPB

The DPB embedding module based on the ACT model proposed in this paper consists of two fully connected layers, with ReLU (Rectified Linear Unit) activation functions applied between the layers for nonlinear transformations. The mathematical model for an input feature vector x with dimension n is as follows:

$$y = W_1 x + b_1, y \in \mathbb{R}^m \tag{1}$$

$$\boldsymbol{y}^{'} = \mathrm{ReLU}(\boldsymbol{y}), \mathrm{ReLU}(\boldsymbol{y}_i) = max(0, y_i), \boldsymbol{y}^{'} \in \mathbb{R}^m \tag{2}$$

$$d = W_2 y^{'} + b_2, d \in \mathbb{R}^m \tag{3}$$

Equations (1), (2), and (3) represent the construction process of the first connection layer, activation function transformation, and final connection layer. Here, $\boldsymbol{W_1}$ and $\boldsymbol{W_2}$ are the weight matrices in the first and final linear layers, with matrix shapes of $m \times n$ and $m \times m$, respectively; $\boldsymbol{b_1}$ and $\boldsymbol{b_2}$ are the bias vectors in the two linear layers, both in the m-dimensional vector space; $\boldsymbol{y}$ and $\boldsymbol{y'}$ are intermediate parameters in the DPB embedding process; d is the dynamic position bias generated from the input features.

To enhance model stability, the weight coefficients are initialized using the Xavier initialization method. At the same time, to improve the adaptability of the weight matrices and bias vectors to the task, a loss function minimization method is employed, and the parameters are updated after each forward and backward pass in the CVAE encoder.

Initialization

Considering the learnability of the bias vectors, they are initialized to zero in the initialization phase, i.e., $\boldsymbol{b_1} = \boldsymbol{b_2} = [0, 0, \cdots, 0]^T$.

The Xavier method is applied to initialize the weight matrices in the first and final connection layers to avoid the impact of gradient explosion and gradient vanishing. For the first layer weight coefficients $\boldsymbol{W_1}$, the input dimension n_{in} and output dimension n_{out} are n and m, respectively; for the final layer weight coefficients $\boldsymbol{W_2}$, both n_{in} and n_{out} are m.

$$W_{ij,initial} \sim N(0, \frac{2}{n_{in}+n_{out}}) \tag{4}$$

Loss Function

Given the need for accurate position perception in robotic fine manipulation, the mean squared error used in the ACT algorithm [9] is retained to measure the difference between the predicted and true values, as shown in equation (5). Here, k and m are the action chunk length and embedding dimension of the ACT model, respectively; $\hat{a}_{t+i,j}$ and $a_{t+i,j}$ represent the predicted and sample action sequences.

$$L = \frac{1}{k} \sum_{i=1}^{k} \sum_{j=1}^{m} \left(\hat{a}_{t+i,j} - a_{t+i,j}\right)^2 \tag{5}$$

Gradient Update

Considering the large dataset of the ACT model and the substantial workload for parameter optimization in the CVAE encoder and decoder, the Stochastic Gradient Descent method is applied to update the weight and bias matrices in the DPB module.

$$\begin{cases} W_i \leftarrow W_i - \eta \frac{\partial L}{\partial W_i} \\ b_i \leftarrow b_i - \eta \frac{\partial L}{\partial b_i} \end{cases} \tag{6}$$

DOI: 10.54254/2755-2721/147/2025.22913

In equation (6), $\boldsymbol{W_i}$ and $\boldsymbol{b_i}$ represent the weight matrices and bias vectors in the first and final linear layers, and η is the learning rate.

Embedding Strategy

$$Attn = Softmax(\frac{QK^T}{\sqrt{d}} + B_{i,j}) \tag{7}$$

Equation (7) gives the mathematical expression of the DPB module as an additional weighted term in the self-attention mechanism, where $Attn$ represents the attention score; Q, K, and V are the query, key, and value matrices in the self–attention module, respectively; and $\sqrt{d}$ is a constant normalization factor. While this embedding method applies to most Transformer architectures, in the context of the ACT model's fine-grained object perception and dynamic position prediction under multi-scale tasks, simply adding it as an extra term in the attention mechanism does not fully leverage the dynamic bias provided by the DPB module, making it difficult to effectively improve task success rates. Therefore, this paper introduces dynamic position bias into the CVAE encoder as a correction term for the action sequence, enabling the model to better capture the relative positional information between the target object and robotic arm. Meanwhile, compared to the correction in the self-attention mechanism, this embedding strategy introduces the bias earlier, allowing it to better integrate into the entire action sequence generation process and making the style variable generated by the CVAE encoder more prominently represent complex dependencies in the task. Based on Zhao et al.'s work [9], the specific embedding process and the improved model structure are shown in Figure 3 and Algorithm 1.

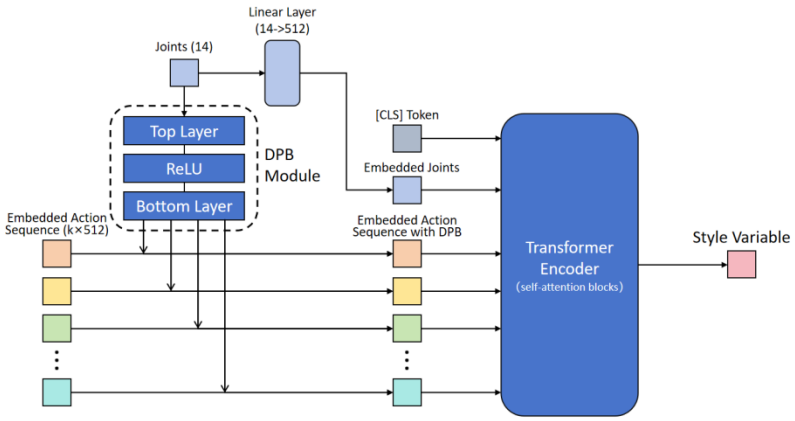


Figure 3: Schematic of the DPB module embedding strategy

Algorithm 1:

1: Retrieve chunk size k, loss weight β, joint positions at t timestep $\boldsymbol{p_t}$, image observations at t timestep $\boldsymbol{m_t}$ and action sequence $\boldsymbol{a_{t:t+k}}$ from the dataset.
2: Initialize CVAE encoder $q_\phi(\boldsymbol{z}|\boldsymbol{a_{t:t+k}}, \boldsymbol{p_t})$ and CVAE decoder $\pi_\theta(\hat{\boldsymbol{a}}_{\boldsymbol{t:t+k}}|\boldsymbol{p_t}, \boldsymbol{m_t}, \boldsymbol{z})$.
3: **for** iteration $n = 1, 2, \cdots$ **do**
4: Input $\boldsymbol{p_t}$ into DPB layer 1, $\boldsymbol{y} = \boldsymbol{W_1 p_t} + \boldsymbol{b_1}$
5: Non-linear transform via ReLU, $\boldsymbol{y}' = \mathrm{ReLU}(\boldsymbol{y})$
6: Input $\boldsymbol{y}'$ into DPB layer 2, $\boldsymbol{d} = \boldsymbol{W_2 y}' + \boldsymbol{b_2}$
7: Add dynamic bias $\boldsymbol{d}$ on $\boldsymbol{a_{t:t+k}}$
8: Obtain $\boldsymbol{z}$ from $q_\phi(\boldsymbol{z}|\boldsymbol{a}'_{\boldsymbol{t:t+k}}, \boldsymbol{p_t})$, $\boldsymbol{a}'_{\boldsymbol{t:t+k}} \leftarrow \boldsymbol{a_{t:t+k}}$
9: Predict $\hat{\boldsymbol{a}}_{\boldsymbol{t:t+k}}$ from $\pi_\theta(\hat{\boldsymbol{a}}_{\boldsymbol{t:t+k}}|\boldsymbol{p_t}, \boldsymbol{m_t}, \boldsymbol{z})$

DOI: 10.54254/2755-2721/147/2025.22913

10: $L_{reconst} = \text{MSE}(\boldsymbol{a}_{t:t+k}, \hat{\boldsymbol{a}}_{t:t+k})$
11: Get L_{re} from ACT policy
12: Update θ, ϕ with ADAM and $L_A = L_{reconst} + \beta L_{re}$; update $\boldsymbol{W_i}$ and $\boldsymbol{b_i}$ with SGD and $L_B = L_{reconst}$
13: **end for**

4. Experimental result

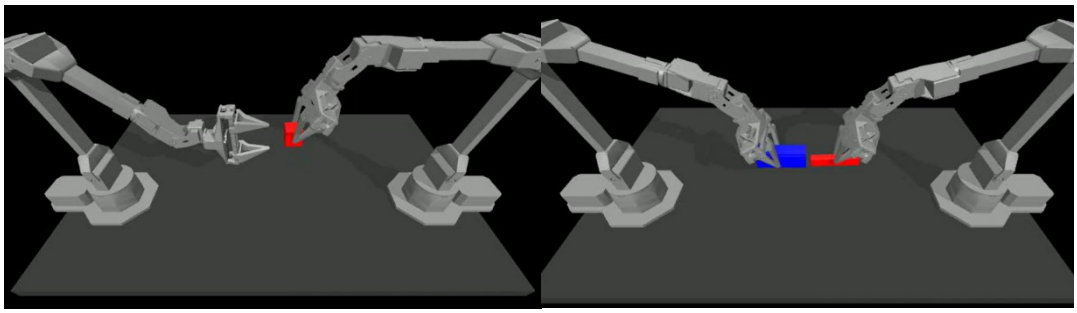
4.1. Model validation

In the simulation experiments, two imitation learning tasks supported by the Aloha environment, "Cube Transfer" and "Bimanual Insertion," were set as the evaluation criteria for the DPB-Embedded ACT model. Their hyperparameters and training process are shown in Table 1 and Figure 5.

- In the Cube Transfer experiment, a red cube is randomly generated within the robotic arm's grasping range. The right-side robotic arm first perceives the position of the object, attempting to touch and grasp the cube in an appropriate posture. Then, the left and right robotic arms work together to transfer the cube to the left-side arm.
- In the Bimanual Insertion experiment, the left and right robotic arms respectively identify and grasp randomly generated slots and rectangular prisms within a specified range. Then, both objects are simultaneously touched and grasped. Finally, the left and right robotic arms work together to attempt inserting the rectangular prism smoothly into the slot.

Table 1. Hyperparameter settings for simulation experiments

	kl_weight	chunk_size	hidden_dim	batch_size	num_epochs	lr
Cube transfer	*10*	*100*	*512*	*8*	*2000*	1×10^{-5}
Bimanual insertion	*8*	*100*	*512*	*8*	*2000*	1×10^{-5}



a) Cube Transfer Task **b) Bimanual Insertion Task**

Figure 5: Robotic arm fine operation simulation experiment

Table 2 presents the success rates of the two fine operation tasks, "Cube Transfer" and "Bimanual Insertion," under various operation strategies. The baseline model data for "BC-ConvMLP" and "BeT" were sourced from the research by Professor Zhao et al. [9]; the data for the "Original ACT" model was obtained from this study's simulation experiments and serves as a reference. The "Refined ACT" refers to the DPB-Embedded ACT model proposed in this paper.

DOI: 10.54254/2755-2721/147/2025.22913

Table 2: Success rates of each model based on two sets of simulation experiments

	Cube Transfer			Bimanual Insertion		
	Touched	Lifted	Transfer	Grasp	Contact	Insert
BC-ConvMLP	34%	17%	1%	5%	1%	1%
BeT	60%	51%	27%	21%	4%	3%
Original ACT	98%	90%	86%	100%	84%	30%
Refined ACT(Ours)	**100%**	**96%**	**90%**	**100%**	**84%**	**42%**

5. Conclusion

This paper addresses the limitations of the ACT model in handling complex action prediction and multi-scale, multi-modal data fusion in fine operation tasks. A DPB embedding strategy based on the ACT model is proposed, which integrates dynamic position bias into the CVAE encoder to enhance positional accuracy, thereby improving task success rates. However, due to the lack of validation on human demonstration datasets, future research will need to further refine the model and evaluate its performance in real-world environments. Future work will focus on reducing the training and inference time of the DPB-Embedded ACT model and expanding its application to more types of tasks. The experimental results show that the proposed DPB-Embedded ACT model performs excellently in task scenarios that require precise position perception and interaction. This progress helps improve robotic arm performance in practical applications such as precision machining, extravehicular activities, and medical surgeries, offering the potential for more efficient and reliable automation solutions.

References

[1] *A. Billard, and D. Kragic, "Trends and challenges in robot manipulation," Science, vol. 364, no. 6446, Jul. 17, 2019. Accessed: Mar. 10, 2025. [Online]. Available: https://www.science.org/doi/10.1126/science.aat8414*

[2] *H. Liu, B. Liang, W. Xu, X. Wang and Y. Shi, "Development of ground experiment system for space robot performing fine manipulation," 2012 12th International Conference on Control Automation Robotics & Vision (ICARCV), Dec. 2012, pp. 1107-1112. doi: 10.1109/ICARCV.2012.6485312.*

[3] *M. Blatnický, J. Dižo, J. Gerlici, M. Sága, T. Lack, E. Kuba, "Design of a robotic manipulator for handling products of automotive industry," International Journal of Advanced Robotic Systems, vol. 2020, no. 1, Jan. 17, 2020, pp. 1-11. doi: 10.1177/1729881420906290.*

[4] *J. Huang, et al., "Comparison of lobectomy performed through Toumai® surgical robot and da Vinci surgical robot in early-stage non-small cell lung cancer: a retrospective study of early perioperative results," Transl Lung Cancer Res, vol. 12, no. 1, pp. 2219-2228, Nov. 30, 2023. doi: 10.21037/tlcr-23-603.*

[5] *C. Ha, S. Jung, J. Park, J. Lim, "Development of a Magnetic Levitation Wafer Handling Robot Transfer System with High-Accuracy and High-Cleanliness: Experimental Evaluation," Applied Sciences, vol. 13, no. 16, 2023. doi: 10.3390/app13169482.*

[6] *R. Taikizawa, Y. Ohmura, and Y. Kuniyoshi, "Gaze-Based Task Decomposition for Robot Manipulation in Imitation Learning," ArXiv, Feb. 1, 2025. Accessed: Mar. 13, 2025. [Online]. Available: https://arxiv.org/html/2501.15071v2*

[7] *M. Siam, A. Singh, C. Perez, C. and M. Jagersand, "4-DoF Tracking for Robot Fine Manipulation Tasks," 14th Conference on Computer and Robot Vision, pp. 329-336, 2017. doi: 10.1109/CRV.2017.41.*

[8] *L. Peternel, N. Tsagarakis, and A. Ajoudani, "A Human-Robot Co-Manipulation Approach Based on Human Sensorimotor Information," IEEE Transactions on Neural Systems and Rehabilitation Engineering, vol. 25, no. 7, pp. 811-822, 2017. doi: 10.1109/TNSRE.2017.2694553.*

[9] *T.Z. Zhao, V. Kumar, S. Levine and C. Finn, "Learning Fine-Grained Bimanual Manipulation with Low-Cost Hardware," ArXiv, Apr. 2023. Accessed: Mar. 13, 2025. [Online]. Available: https://arxiv.org/abs/2304.13705*

[10] *M. Kobayashi, T. Buamanee, and T. Kobayashi, "Alpha-α and Bi-ACT Are All You Need: Importance of Position and Force Information/ Control for Imitation Learning of Unimanual and Bimanual Robotic Manipulation With Low-Cost System," IEEE Access, vol. 13, pp. 29886-29899, 2025. doi: 10.1109/ACCESS.2025.3541200.*

[11] J.H. Park, et al., "Hierarchical Action Chunking Transformer: Learning Temporal Multimodality from Demonstrations with Fast Imitation Behavior," 2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 12648-12654, 2024. doi: 10.1109/IROS58592.2024.10802845.
[12] Q. Zhou, H. Zou and H. Wu, "LGViT: A Local and Global Vision Transformer with Dynamic Contextual Position Bias Using Overlapping Windows," Applied Sciences, vol. 13, no. 3, 2023. doi: 10.3390/app13031993.
[13] S.A. Værnø, A.H. Brodtkorb and R. Skjetne, "Compensation of Bias Loads in Dynamic Positioning of Marine Surface Vessels," Ocean Engineering, vol. 178, pp. 484-492, 2019. doi: 10.1016/j.oceaneng.2019.03.010.
[14] R. Sheh, B. Hengst and C. Sammut, "Behavioural Cloning for Driving Robots over Rough Terrain," 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 732-737, 2011. doi: 10.1109/IROS.2011.6094652.
[15] F. Hsiao, J.H. Kuo and M. Sun, "Learning a Multi-Modal Policy via Imitating Demonstrations with Mixed Behaviors," ArXiv, Mar. 25, 2019. Accessed: Mar. 13, 2025. [Online]. Available: https://arxiv.org/abs/1903.10304
[16] C. Fei, et al., "Triple-GAIL: A Multi-Modal Imitation Learning Framework with Generative Adversarial Nets," ArXiv, May. 22, 2020. Accessed: Mar. 17, 2025. [Online]. Available: https://arxiv.org/abs/2005.10622
[17] N.M.M. Shafiullah, Z.J. Cui, A. Altanzaya and L. Pinto, "Behavior Transformers: Cloning k modes with one stone," ArXiv, Oct. 11, 2022. Accessed: Mar. 18, 2025. [Online]. Available: https://arxiv.org/abs/2206.11251
[18] K. Sohn, X. Yan and H. Lee, "Learning Structured Output Representation using Deep Conditional Generative Models," Proceedings of the 29th International Conference on Neural Information Processing Systems, vol. 2, pp. 3483-3491, Dec. 7, 2015. doi: 10.5555/2969442.2969628.
[19] B. Ivanovic, K. Leung, E. Schmerling and M. Pavone, "Multimodal Deep Generative models for trajectory Prediction: A Conditional Variational Autoencoder Approach," IEEE Robotics and Automation Letters, vol. 6, no. 2, pp. 295-302, Apr. 2021. doi: 10.1109/LRA.2020.3043163.
[20] Y. Zhang, J. Su, H. Guo, C. Li, P. Lv and M. Xu, "S-CVAE: Stacked CVAE for Trajectory Prediction with Incremental Greedy Region," IEEE Transactions on Intelligent Transportation Systems, vol. 25, no. 12, pp. 20351-20363, Dec. 2024. doi: 10.1109/TITS.2024.3465836.
[21] H. Zhou, X. Yang, D. Ren, H. Huang and M. Fan, "CSIR: Cascaded Sliding CVAEs With Iterative Socially-Aware Rethinking for Trajectory Prediction," IEEE Transactions on Intelligent Transportation Systems, vol. 24, no. 12, pp. 14957-14969, Dec. 2023. doi: 10.1109/TITS.2023.3300730.
[22] Wang, et al., "CrossFormer: A Versatile Vision Transformer Hinging on Cross-Scale Attention," ArXiv, Oct. 8, 2021. Accessed: Mar. 17, 2025. [Online]. Available: https://arxiv.org/abs/2108.00154
[23] W. Zhang, Q. Chai, Q. Zhang, C. Wu, "Obstacle-transformer: A trajectory prediction network based on surrounding trajectories," Terrain-Vehicle System: Modelling, Perception and Intelligent Control, vol. 5, no. 1, Mar. 2023. doi: https://doi.org/10.1049/csy2.12066.
[24] W. Wang, et al., "CrossFormer++: A Versatile Vision Transformer Hinging on Cross-Scale Attention," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 46, no. 5, pp. 3123-3136, May. 2024. doi: 10.1109/TPAMI.2023.3341806.