Frontiers in Computing and Intelligent Systems
ISSN: 2832-6024 | Vol. 8, No. 3, 2024

Contrastive Prediction and Estimation of Deformable Objects based on Improved Resnet

Haipeng Gao [a], Yadong Teng [b]

School of Computer Science and Technology, Qingdao University, Qingdao, Shandong, China
[a] 2209309768@qq.com, [b] 1912741882@qq.com

Abstract: Because the dynamic model of deformable linear object is complex, the learning based on visual model is difficult, and the feature information extraction is insufficient. Therefore, we propose a joint visual representation model using contrast learning of optimized encoder. We start with the encoder, add the residual structure to the encoder, optimize the extraction and compression of its feature information, and control its parameters to 3 million. In this way, we can not only obtain excellent feature information, but also have good efficiency. In the rope experiment, we collect information from the simulated environment without manual marking, extract features through the encoder and transmit them to the downstream task. Experiments show that the evaluation of our model at 135 ° and 45 ° is improved by about 50%.

Keywords: Contrastive Prediction; Resnet; Deformable.

1. Introduction

In the past few decades, robot arm manipulation has developed rapidly, and has made great progress in many subdivided fields. Rigid object manipulation is extremely mature. Different from rigid objects, the dynamic model of deformable objects is complex and there is no standardized state, so the operation of deformable objects is still a challenging task, and the control of deformable linear objects (DLO) is one of the representative problems. The manipulation of deformable objects has a wide range of applications, such as robotic surgery, assisted dressing, cable routing, folding clothes, and threading in the textile industry [1, 2, 3, 4, 5, 6, 7, 8]. However, deformable objects have complex and nonlinear dynamics with high degrees of freedom, which makes state estimation challenging and training prediction expensive.

Using the robotic arm, placing the rope from a random initial state to the target state, or using a two-arm robot to tie the rope, all of which require the robot to observe and generate corresponding actions to operate according to the current state of the rope. The dynamic model of deformable objects is complex and nonlinear, and even simple objects face complex and unpredictable behavior. We choose the learning based method among many methods, but if there is no excellent model, its effect is general and generalization (the purpose of learning is to learn the laws hidden behind the data, and the trained network can also give appropriate output for the data other than the learning set with the same law) is weak.[9, 10, 11] With the introduction and improvement of elf-supervised learning, elf-supervised learning directly uses the data itself to provide supervision information to guide learning, so its training cost is reduced. The emergence of contrast learning method based on self supervised learning not only reduces the training cost of images and labels, but also can extract a general and easy to transform feature model. In the shape control of DLO, the deformable object prediction representation learning proposed by Wilson Yan [12] et al. based on contrast estimation is feasible. However, due to its predictive control from various angles, its final state is easy to have a large difference from the target state. We speculate that the extraction of feature information is not perfect and the depth of the model is not enough. T Therefore, on this basis, a new network structure based on comparative learning is proposed to deal with deformable linear objects, improve the ability of feature information extraction and optimize the depth of the model.

In terms of DLO operation, the requirements for accuracy are also increasing. In this paper, with the popularity of many efficient frameworks and the open source of various network structures, we learn a variety of frameworks and train a new framework based on visual model. The framework jointly learns visual representations of latent spaces and dynamic models of deformable objects using contrastive optimization. Inspired by SimCLR [13] and Resnet [14], a special deep convolutional network with a separable residual network is added to the extraction of features from the encoder to the latent space, which not only improves the efficiency of feature extraction and quality, and balances the speed and quality of feature extraction. The residual network structure can be added to most neural networks to effectively solve the problems such as gradient explosion, learn effective model dynamics after feature extraction, and use standard Model Predictive Control (MPC) and one-step prediction to operate deformable objects and let them reach the target position. In experiments, we have made considerable progress compared with other baseline methods.

In summary, the main contributions of this paper can be summarized as follows:

(a) We propose a DLO shape control compatible model predictive control method based on comparative learning

(b) After modifying the network structure, compared with other baseline methods, we find that our method has the potential representation of stronger learning and more planning, and its feature information extraction has more advantages

In this paper, we detail the shape control of deformable objects based on a contrastive learning method with an improved network structure, and we organize the rest of this paper as follows. In Section II, were view the work related to our method. In the third part, we present our model in detail, including a detailed description of some of the components used in the method, as well as the mathematical formulations and parameters used in the code implementation. In Section IV, we have carried out experiments on a data set made of collected data in the simulation environment, and compared it with some baseline methods to show that our method is relatively advanced. At the end of this section, we demonstrate the effectiveness of the improved module through ablation experiments using a control variable

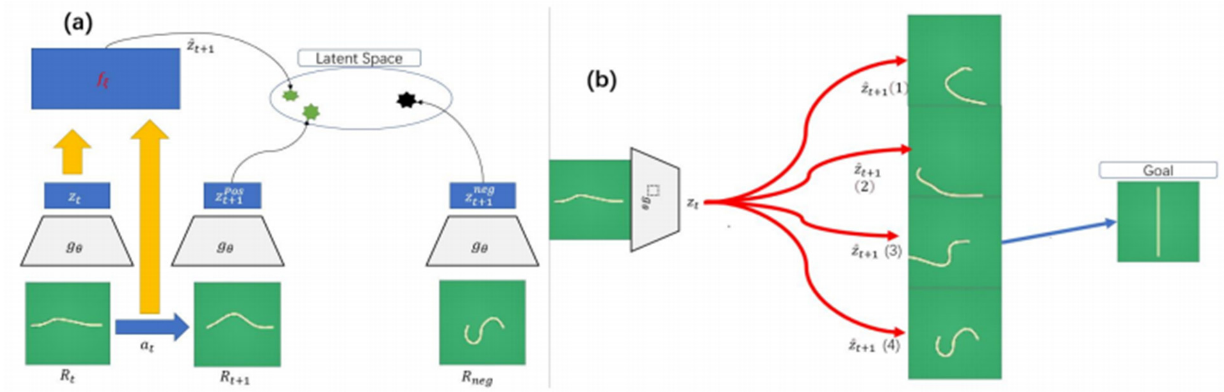

Figure 1. Comparative learning process

(a): The training data consists of (image, next image, action), the encoder, and forward model parts of the model we designed. The optimized contrastive loss objective makes positive embedding pairs closer and negative embeddings farther.

(b): We run our model by designing multiple actions, and applying different actions. Ultimately, we choose a set that is closer to our target model approach. Section V summarizes our work and future research prospects.

2. Related Work

2.1. Deformable Object Manipulation

Robotic manipulation of deformable objects has a rich history spanning diverse domains, from industrial operations to everyday life to surgical robotics. Manipulation of Deformable Linear Objects there has been a lot of previous work in the field of robotic manipulation of deformable objects. For detailed information, the reader is referred to Khalil and Payeur[15], Henrich and Wörn[2]. The standard approach to linear manipulations such as ropes is to use deformable object simulation in conjunction with planning methods [16]. Past work in this area has focused on simple linear deformable objects [17, 18, 19], creating better simulations [20], and faster planning [21]. Rodriguez et al. [21] developed methods for deformable simulation environments, and Frank et al. developed methods for faster planning in deformable environments. But the various states of deformable linear objects are difficult to plan correctly even with high computational efficiency. Some previous methods deal with complex dynamics through local controllers, rather than planning complete complex dynamics focusing on simpler approximate programming. One way to use local controllers is model-based serv serving [22], where the end- effector is controlled to the target position rather than being explicitly planned [23]. However, since controllers are optimized on simple dynamics, they often get stuck in local minima for more complex dynamics [24]. To address this model-based dependency, Berenson [25], McConachie and Berenson [26], Navarro-Alarcon et al. [27] investigated Jacobian approximation controllers that do not require explicit models, while Jia et al. Ren [28], Hu et al. [29] studied the learning-based servo technique. However, since controllers are still local, they are still prone to global suboptimal policies. To address this problem McCain's McConachie et al. [24] combined planning with a local controller. Although this results in better behavior, transferring it to robots requires solving difficult state estimation problems [3, 4], however, these will have various accuracy and efficiency issues, we propose a new and improved training potential dynamics model based on comparative learning. The model uses the framework of end-to-end comparative learning to propose an information encoder model, which improves the ability of feature information extraction, so as to improve the accuracy of rope at all angles.

2.2. Contrastive Learning

The learning of dynamic models and the extraction of well-representative information remains a formidable challenge in deformable linear object manipulation. Contrastive learning is developing rapidly, and a lot of research has been done to better represent data. Word2Vec optimizes a contrastive loss to demonstrate semantic and syntactic structure in the latent space of word learning. [30] showed that high-level representations for image, video and speech data can be learned by using a large number of negative samples. Tian et al[31] learn high-level representations through a framework-like contrast loss by keeping positive samples of different scenes close to each other and far away from other scenes. The new contrastive learning framework simclr [13] has made important progress in feature representation, providing a new method for developing flexible linear object capture. This method combines different samples in the embedded space, so that the positive samples should be as close as possible and the negative samples should be as far away as possible. In end-to-end learning, one encoder generates positive samples and the other generates negative samples for training and learning, and then passes them to downstream tasks. This method costs more batch size. We designed a model of a single encoder, and in the encoder stage, we combined the residual structure to optimize the structure of each step, so that the batch size does not need to be particularly large. We set it to 128, and the extraction of feature information has made great progress, so that it can basically reach the required shape in the evaluation stage.

3. Proposed Method

In this section, we describe the framework we trained for deformable linear object manipulation: forward modeling based on contrastive learning. In this section, we first discuss the predictive models of contrastive learning, and important improvements to the models inspired by Resnet, and secondly, we discuss the formalism of predictive modeling and contrastive learning. For our training program, please refer to Figure 1.

3.1. Predictive Models

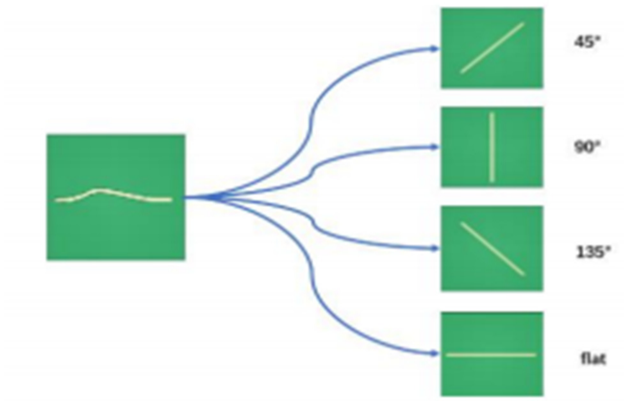

Figure 2. From the initial state, through our model to reach different target states, we select several classical target states as levels respectively. 45°. 90°. 135°

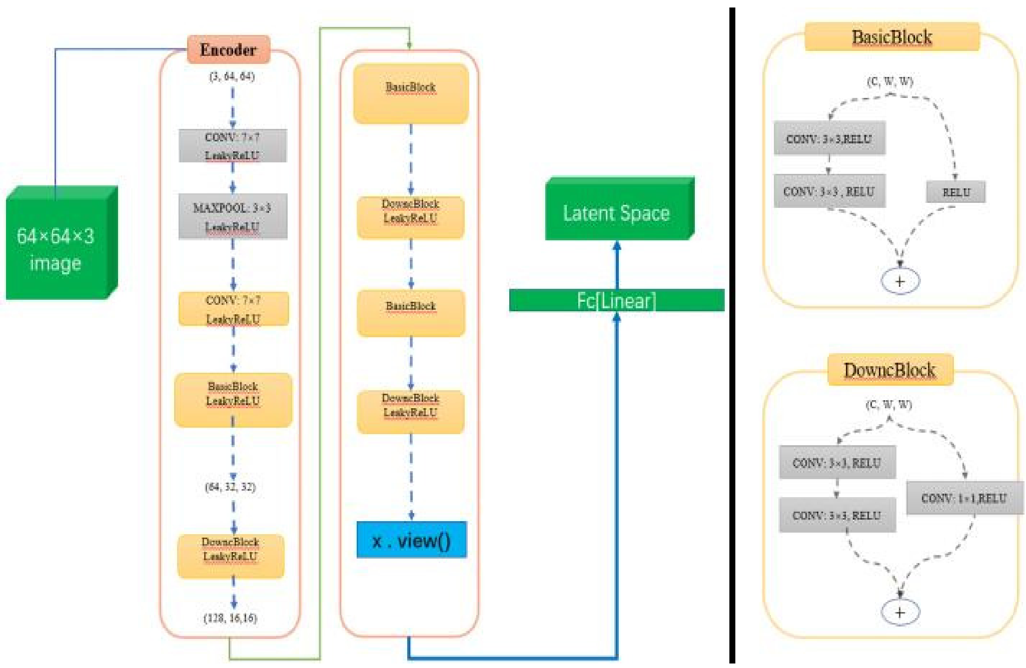

Figure 3. This is the encoder structure we designed, which contains two residual structures. The image is input and extracted by the encoder and compressed into the potential space.

For the formulation of the problem, an environment is set where observations $o \in O$, actions $a \in A$, and deterministic transition dynamics $f(o_t, a_t) = o_{t+1}$. This can learn to train latent features through pixel latent space to predict the action of the next time step, and obtain observation-action-observation tuples [33] through observation. When training a prediction model, we can use it to plan deformable linear objects in different states to reach different expected target states. For example, please refer to Figure 2 for the different operations of the rope. If the planning is done through the pixel space of the picture, that is, the pixel value of the picture, there will be a great precision deviation, because the distance between the pixel values of the picture cannot have a good correlation with the real distance. For example, in a task where the screw is placed in the center of the box, the screw needs to be placed in the center of the hole. If the screw is farther from the center of the box, then if comparing in pixel space, the next action may be the same distance from the center image when using the predictions of the vision-forward model, since there is no image overlap here. Therefore, the prediction framework we consider is to compress the image in the latent space for training and prediction, and then train and predict through the encoder. We design a Resnet-based encoder, the network structure is shown in Figure 3, the training data is embedded in a latent space, and a forward prediction model is combined in the latent space. Contrastive Methods learn through positive and negative samples, so we found that end-to-end contrastive learning is more suitable for learning the latent space, in which the design of the encoder part is particularly important, which determines the efficiency of training and the success of downstream tasks.

3.2. Contrastive Learning

In the overall process of contrastive learning, first collect data and make data sets. Then press the data into the latent space. This is the most important step. Only by designing the encoder well can the overall efficiency and data compression quality be guaranteed, as shown in Figure 3. Finally, the data of the positive and negative samples are compared. The gradient calculation is performed by comparing the loss function so that the embedding of positive samples is closer, and the embedding of negative samples is farther. We design the encoder assuming $g_\theta(o_t) = z_t$ and the forward model $f\varphi(z_t, a_t) \approx z_{t+1}$, using the InfoNce contrastive loss described by Oord et al [37] InfoNce contrastive loss $L_q = -\log\left(\frac{\exp(q \cdot \frac{k_+}{\tau})}{\sum_{i=0}^{K}\exp(q \cdot \frac{k_i}{\tau})}\right)$ (1) q represents the data to be checked, and k_+ represents the sample similar to q. The numerator is the similarity of positive samples, and the denominator is the similarity of positive samples and all negative samples. The minimized InfoNCEloss is to maximize the numerator and minimize the denominator, that is, to minimize the similarity of positive samples and the similarity of negative samples. Then we use the loss proposed by Wilson Yan [12] et al. $L = -E_D\left[\log \frac{h(\hat{z}_{t+1}, z_{t+1})}{\sum_{i=1}^{k} h(\hat{z}_{t+1}, \hat{z}_i)}\right]$ (2) On this basis, where h is the similarity function between the computed embeddings, ẑt+1 represents the next state of the negative sample, and k represents that there are k such negative samples. The essence of contrastive learning is to separate the positive and negative samples, the positive samples are together, and the negative samples are further separated, as shown in Figure 1. Then we need to learn a minimal forward model $\|f\varphi(z_t, a_t) - z_{t+1}\|$(3) The similar function used is $h(z_1, z_2) = \exp(-\|\lambda_1 - z_2\|^2)$(4) After learning the encoder we enter the action stage of the objective function, that is, the realization of the downstream task. Here we also need to use a simple model predictive control (MPC), according to the type of our action Select az_t forward model to run, and select the next action that is closest to the target within the distance of (4) the similarity function, which also completes the operation of the deformable linear object based on contrastive learning. In the feature extraction, the encoder that compresses the data into the latent space is an important link between the previous and the next.

3.3. Encoder

We add domain randomization to data processing, generate a large number of different simulation scenes, extract better feature information, and make the network adapt to different domains. In contrast learning, the most important part is feature extraction. Only good enough feature information can help subsequent tasks. In the network proposed by Wilson Yan [12], only six layers are implemented. Simple convolution network has low feature dimension and strong local information, but it lacks the diversity of feature information. However, the network with too deep depth is easy to lead to gradient dispersion, while the simple depth network is more prone to feature redundancy, and only a small part of image features is extracted. The result of adding the remaining network structure has changed significantly. We designed a new encoder structure, as shown in Figure 3. It is mainly composed of two residual structures: RestNetBasicBlock and RestNetDownBlock. The difference between the two remaining structures is that in the final output accumulation stage, our RestNetDownBlockis increased by 1 × 1. This can reduce the number of parameters in the deep network and improve efficiency. The joint effect of the two structures is to deepen the network structure and improve the accuracy. At the same time, it can solve the problem of network degradation, so as to achieve a more efficient feature extraction. 64 ×64× 64 images are input into our encoder, first through a layer of convolution kernel with a size of 7 × 7. The step size is 1 and the Paddingis 3, which can better focus the required feature information. Then through Maxpool and the two residual structures we designed, and finally through linearization, the output data is added to the potential space. The encoder we designed reduces the number of parameters of Resnet network to 3.8 million. The number of parameters of the original resnet18 network far exceeds this number, reaching the level oftens of millions. LeakyReLu [27] is added to the neural network to introduce nonlinearity. At the same time, it solves the problem of Relu neuron death. It improves the efficiency and diversity of feature extraction.

4. Experiment and Evaluation

In this section, we will conduct experimental evaluation of our method and simulate different rope operation tasks in different environments. We will compare different models on a unified dataset to prove that our method really works. It also addresses the question of whether contrastive learning for improved encoders has a better potential for manipulation of deformable linear objects. Later in this section, we will also conduct ablation experiments of separate modules on the same dataset, demonstrating the layer-by-layer progress of our method.

4.1. Data Collection

The Deep Mind Control [33] platform of the MuJoCo [34] engine we use uses a 64×64×3 RGB image as input. The deformable linear object we designed is a rope composed of 25 geometric bodies, as shown in Figure 4. Supervised learning is expensive and time-consuming to collect samples for manual labeling, while comparative learning like self-supervised learning does not require manual labeling information, and directly uses the data itself as supervision information to learn the feature expression of sample data and use it as a downstream task, then we solve this problem by perturbing the rope through random actions in the simulation environment. We collected 20,000 rope trajectories with a length of 10 (200k samples). Part of the rope with different trajectories.

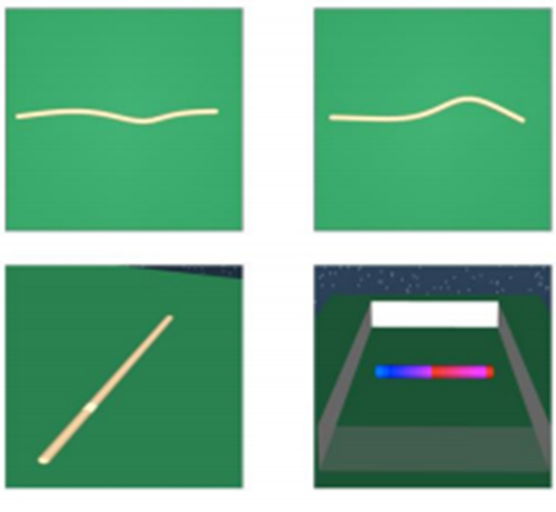

Figure 4. The first two figures are generated pictures, and the last two pictures are 25 geom simulation environment ropes

4.2. Baselines Training

We employ several methods to compare our improvements, a visual forward model[38], an autoencoder trained jointly with a latent dynamics model[37], and PlaNet [36]. and Wilson Yai's unimproved contrastive learning-based control prediction [12]. To ensure that the action point is on the rope lock, we use RGB thresholds to limit its action point, and the MPC method of one- step prediction is used in these methods.

- Visual Forward Model: Kaise's forward model is used to model the pixel space to control the prediction.
- Autoencoder: Compress the model trained on the latent space via a classical autoencoder and a simple forward dynamics model.
- PlaNet: By simulating a VAE variational autoencoder, the training and learning of the latent space are performed, and the temporal variation downline is optimized.
- CFM: Based on the sample processing of contrastive learning, the latent space is trained and learned through a simple 6-layer convolutional encoder.

We have the same environment in all methods, and all ropes have the same number and length. We collected the predictions of all its methods to compare with our method and the predicted trajectory of each method.

4.3. Training

The encoder we use is a convolutional neural network with a residual structure. The encoder first goes through a two- dimensional convolution with a kernel size of 7 and a stride of 1. After the maximum pooling layer, the amount of computation is reduced to remove redundant structures and reduce Our parameters are then added to the RestNetBasicBlock and RestNetDownBlock layers, and LeakyReLU [27] activations are added between each layer, and the final output is flattened and passed into the fully linked layer. Our kernel size [7,3,1,3,3,3,3,1,3,3, 3,3,1,3, 3,3,3,1] per layer for our method, we use contrastive learning with 127 negative samples per batch. The learning rate used is 1e-3, the weight_decay is 1e-6, the optimizer is Adam [21], and 30 epochs are trained. All methods are trained on Intel® Xeon(R) W-2102 CPU @ 2.90GHz × 4 and GeForce GTX 1050 Ti.

4.4. Quantitative Assessment

In this section, we quantitatively evaluate our method against baseline methods. The training results of all our methods are shown in Table 1. The evaluation unit used for the evaluation of all our methods is geom, which represents the sum of the geometric distance between the final result and the target result. It can be found that our proposed method has obvious advantages in different angles, and the error is the smallest in horizontal angles. This shows that our method has better generalization ability and better performance in the latent space, which proves that our method is indeed effective and has progressed. All methods run 20 actions in the same environment with the same initial state and target state, and our method first reaches the target state. As shown Figure 5.

4.5. Ablation Experiment

In this section, we will perform ablation experiments on network structures and datasets to demonstrate the superiority of our method.

(1) In the first experiment, we will replace the simple convolutional network in the encoder with a new network with our improved residual network structure. The original simple convolutional network is a network with only six layers of two-dimensional convolution and Relu activation functions. The complexity of the simple network is low, and the extracted information is not perfect. And our network has more advantages in feature information extraction. The training results are shown in Table 3.

(2) In the second experiment, we will discuss the dataset, adding domain randomization, the rest being the same. After adding this operation, the data set samples have a variety of changes, which will be more suitable for the display environment, and it can be seen in the simulation test that there is indeed progress. The experimental results are shown in Table 2.

Table 1. Quantitative comparison between different methods of manipulation in the rope task. The evaluation unit is the geometric distance (Geom) between the final task and the target state. The smaller the distance, the better.

Rope					
method	90。	0。	135。	45。	Random
Visual Forward Model	2.38	2.09	2.10	2.29	1.52
Autoencoder	1.96	1.72	1.85	2.11	4.30
PlaNet	2.58	1.81	2.18	2.31	3.03
CFM	1.89	0.58	1.34	2.29	1.52
Our	1.07	0.48	0.62	1.01	1.40

Table 2. The data set has undergone Domain Randomization to adapt it to more environments. In quantity evaluation of various methods on its data set, the evaluation criteria are the same as those in Table 1.

Rope with DR					
method	90。	0。	135。	45。	Random
Visual Forward Model	5.28	2.60	4.07	4.47	4.59
Autoencoder	2.23	1.79	2.16	2.05	3.16
PlaNet	2.22	1.79	2.12	2.07	2.80
CFM	1.13	0.68	0.82	0.92	1.38
Our	0.83	0.47	0.51	0.73	1.37

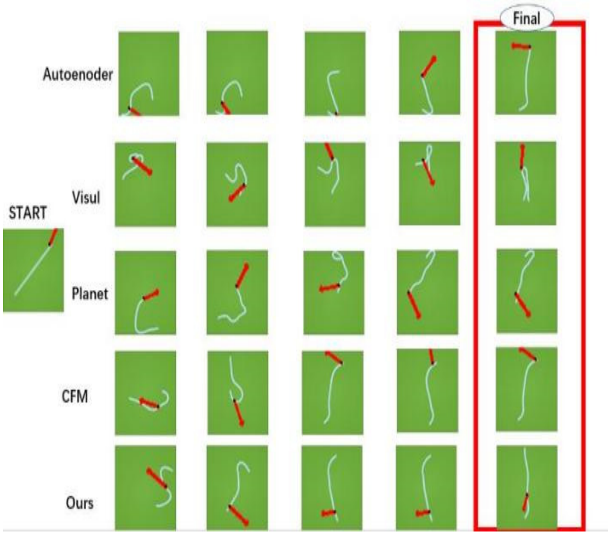

Figure 5. All methods run 20 actions whose target state is the rope vertical, and we sample every 4 actions. Obviously our method reaches the target state first

Table 3. Regarding the ablation experiments of the encoder, all the criteria are the same, only the network model of the encoder is different, and the evaluation criteria are the same as Table 1.

method	90。	0。	135。	45。	Random
simple	1.8	05	1.3	2.2	1.5
ours	1.0	0.4	0.6	1.0	1.4

5. Conclusion

In this paper, we focus on modifying the encoder network structure, deepening the network structure and avoiding network degradation and over fitting. Under the condition of appropriate parameters, it greatly improves the extraction ability of feature information, obtains more effective feature information, and makes the final state of rope shape closer to the target state. In the future work, we will do more experiments on the expansion of deformable objects, from ropes to more deformable flexible objects, so that they can achieve more types of operation tasks.

References

[1] Takahiro Wada, Shinichi Hirai, Sadao Kawamura, and Norimasa Kamiji. Robust manipulation of deformable objects by simple positive feedback. In ICRA, 2001.

[2] Dominik Henrich and Heinz Wörn. Robot manipulation of deformable objects. In Springer Science & Business Media, 2012.

[3] John Schulman, Jonathan Ho, Cameron Lee, and Pieter Abbeel. Generalization in robotic manipulation through the use of non-rigid registration. In ISRR, 2013.

[4] John Schulman, Alex Lee, Jonathan Ho, and Pieter Abbeel. Tracking deformable objects with point clouds. In ICRA, 2013.

[5] Yilin Wu, Wilson Yan, Thanard Kurutach, Lerrel Pinto, and Pieter Abbeel Learning to manipulate deformable objects without demonstrations. In arXiv preprint, 2019.

[6] Daniel Seita, Aditya Ganapathi, Ryan Hoque, Minho Hwang, Edward Cen, Ajay Kumar Tanwani, Ashwin Balakrishna, Brijen Thananjeyan, Jeffrey Ichnowski, Nawid Jamali, Katsu Yamane, Soshi Iba, John Canny, and Ken Goldberg. Deep imitation learning of sequential fabric smoothing policies. In arXiv preprint,2019.

[7] Jeremy Martin-Shepard, Marco Cusumano-Towner, Jinna Lei, and Pieter Abbeel. Cloth grasp point detection based on multiple-view geometric cues with application to robotic towel folding. In ICRA, 2010.

[8] Jan Stria, Daniel Prusa, V aclav Hlavac, Libor Wagner, Vladimir Petrik, Pavel Krsek, and Vladimir Smutny. Garment perception and its folding using a dual-arm robot. In IROS, 2014.

[9] Tuomas Haarnoja, Aurick Zhou, Kristian Hartikainen, George Tucker, Sehoon Ha, Jie Tan, Vikash Kumar, Henry Zhu, Abhishek Gupta, Pieter Abbeel. Soft actor-critic algorithms, and applications. In arXiv preprint, 2018.

[10] John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. Trust region policy optimization. In ICML,2015.

[11] Timothy PLillicrap , Jonathan J Hunt, Alexander Pritzel , Nicolas Heess, Tom Erez , Yuval Tassa , David Silver , and Daan Wierstra . Continuous control with deep reinforcement learning. In arXiv preprint, 2015.

[12] Wilson Yan, Ashwin V angipuram, Pieter Abbeel, and Lerrel Pinto, Learning Predictive Representations for Deformable Objects Using Contrastive Estimation 2020.

[13] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. arXiv preprint, 2020.

[14] He K ,Zhang X ,Ren S. Deep Residual Learning for Image Recognition[J].IEEE, 2016.

[15] Fouad F Khalil and Pierre Payeur. Dexterous robotic manipulation of deformable objects with the multi-sensory feedback-a review. In Robot Manipulators Trends and Development. 2010. Infogan. In NeurIPS, 2018.

[16] P Jiménez. Survey on model-based manipulation planning of deformable objects. Robotics and computer-integrated manufacturing, 2012.

[17] Mitul Saha and Pekka Isto. Manipulation planning for deformable linear objects. In T-RO, 2007.

[18] Hidefumi Wakamatsu, Eiji Arai, and Shinichi Hirai. Knotting/unknotting manipulation of deformable linear objects. IJRR, 2006.

[19] Mark Moll and Lydia E Kavraki. Path planning for deformable linear objects. T-RO, 2006.

[20] Samuel Rodriguez, Xinyu Tang, Jyh-Ming Lien, and Nancy M Amato. An obstacle-based rapidly-exploring random tree. In ICRA, 2006.

[21] Barbara Frank, Cyrill Stachniss, Nichola Abdo, and Wolfram Burgard. Efficient motion planning for manipulation robots in environments with deformable objects. In IROS, 2011.

[22] Grady Williams, Nolan Wagener, Brian Goldfain, Paul Drews, James M Rehg, Byron Boots, and Evangelos A Theodorou. Information-theoretic Mpc for model-based reinforcement learning. In ICRA, 2017.

[23] Liangpeng Zhang, Ke Tang, Xin Yao.Explicit Planning for Efficient Exploration in Reinforcement Learning-NeurIPS 2019.

[24] Dale McConachie, Mengyao Ruan, and Dmitry Berenson. Interleaving Planning and control or deformable object manipulation. In International Symposium on Robotics Research (ISRR), 2017.

[25] Dmitry Berenson. Manipulation of deformable objects without modeling and simulating deformation. In 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems pages 4525–4532. IEEE, 2013.

[26] Dale McConachie and Dmitry Berenson. Estimating model utility for deformable object manipulation using multiarmed bandit methods. IEEE Transactions on Automation Science and Engineering, 15(3):967–979, 2018.

[27] David Navarro-Alarcon, Y un-hui Liu, Jose Guadalupe Romero, and Peng Li. On the visual deformation serving of compliant objects:Uncalibrated control methods and experiments. The International Journal of Robotics Research, 33(11):1462– 1480, 2014.

[28] Nabil Essahbi, Belhassen Chedli Bouzgarrou, and Grigore Gogu. Soft material modeling for robotic manipulation. In Applied Mechanics and Materials, Volume 162, pages 184–193. Trans Tech Publ, 2012.

[29] Chelsea Finn and Sergey Levine. Deep visual foresight for planning robotmotion. In 2017 IEEE International Conference on Robotics and Automation (ICRA), pages 2786–2793. IEEE, 2017.

[30] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. In arXiv preprint, 2018.

[31] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. arXiv preprint, 2019.

[32] Josh Tobin, Rachel Fong, Alex Ray, Jonas Schneider, Wojciech Zaremba, Pieter Abbeel Domain Randomization for Transferring Deep Neural Networks from Simulation to the Real World In IROS 2017.

[33] Kaiser, L. , Babaeizadeh, M. , Milos, P. , Osinski, B. , Campbell, R.H. , & Czechowski, K.. Model-based reinforcement learning for atari.Yuval Tassa, Yotam Doron, Alistair Muldal, Tom Erez, YazheLi, Diego de Las Casas, David Budden, Abbas Abdolmaleki, Josh Merel, Andrew Lefrancq. Deepmind control suite. In arXiv preprint, 2018.

[34] Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In IROS, 2012.

[35] Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. Learning latent dynamics for planning from pixels. arXiv preprint, 2018.

[36] Sascha Lange and Martin Riedmiller. Deep auto-encoder neural networks in reinforcement learning. In IJCNN, 2010

[37] Lukasz Kaiser, Mohammad Babaeizadeh, Piotr Milos, Blazej Osinski,Roy H Campbell, Konrad Czechowski, Dumitru Erhan, Chelsea Finn, Piotr Kozakowski, Sergey Levine. Model-based reinforcement learning for atari. In arXiv preprint, 2019.