2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

RoboHitch: Learning Visual Affordance from Disordered Keypoints for Hitch Knots Tying

Jiahui Zuo[1], Boyang Zhang[1], and Fumin Zhang[1], *Fellow, IEEE*

***Abstract*— Robotic manipulation of deformable linear objects (DLOs) presents significant challenges due to complex dynamics and frequent self-occlusions. Existing robotic knot tying methods typically rely on precise topological state tracking with ordered keypoints and explicit edge connectivity. This reliance makes them prone to failures due to tracking drift and topology mismatch caused by repeated bending and crossings during knot formation. To address these limitations, we introduce RoboHitch, a novel framework that learns to perform hitch knot tying from human demonstrations using only disordered 3D keypoints and RGB images. This eliminates the need for explicit topological order, allowing for more flexible manipulation. Our method employs a dynamic Graph Autoencoder to extract geometric features from untracked keypoints, complemented by a Convolutional Autoencoder that captures essential visual context. A bidirectional cross-attention mechanism then fuses these modalities to jointly predict pick and place affordances, facilitating implicit reasoning about the rope's state and enabling knot tying under occlusion. Real-world experiments demonstrate the effectiveness and generalizability of our approach, successfully completing hitch knots in scenarios with self-occlusions.**

I. Introduction

Knotting manipulation of DLOs is widely employed in diverse applications, including surgery, industrial automation, and maritime operations. Knot tying can be classified into three distinct categories: (1) knots, formed within a single rope like overhand knots, (2) bends, connecting between two or more ropes, and (3) hitches, tying a rope to another object. In real-world scenarios, knot-tying is typically performed to fasten a rope to a target object rather than creating an isolated knot. Consequently, this work focuses on robotic hitch knot tying, a task that humans perform with ease, but remains difficult for robotics.

A fundamental challenge in robotic knot tying is DLO state representation. Unlike rigid objects, there is no direct representation of the DLO state [1]. Although RGB images and point clouds are easy to obtain, they are relatively unstructured and often data hungry. To facilitate subsequent motion planning, DLOs are typically represented as ordered lists of keypoints, where the order encodes topology along the rope. This structured prior is useful, but accurate keypoint estimation is difficult during knot formation because self-occlusion can break correspondence and ordering. While previous work has made progress in tracking DLO topology using point cloud registration for overhand knots or simpler configurations [2]–[4], these methods often assume uncrossed DLO initialization and struggle to maintain accurate keypoint tracking from complex self-occlusion states for knot tying [5], [6]. Topology mismatch from tracking failure can easily lead to task failure caused by failed grasps. With a DLO state estimate, both model-based [3], [7], [8] and learning-based methods [9]–[11] seek the optimal action sequences to complete the task. Model-based pipelines typically require explicit and consistent keypoint ordering (often via markers) to classify knot states [12] and execute hand-designed primitives to transition states. The reliance on explicit topological states making them sensitive to occlusion-induced tracking errors without markers. Furthermore, many learning-based methods use RGB or point cloud inputs from easier state estimation and adopt end-to-end frameworks that tightly couple perception and action. This coupling often leads to large quantities of real-world data requirements and makes policies difficult to interpret.

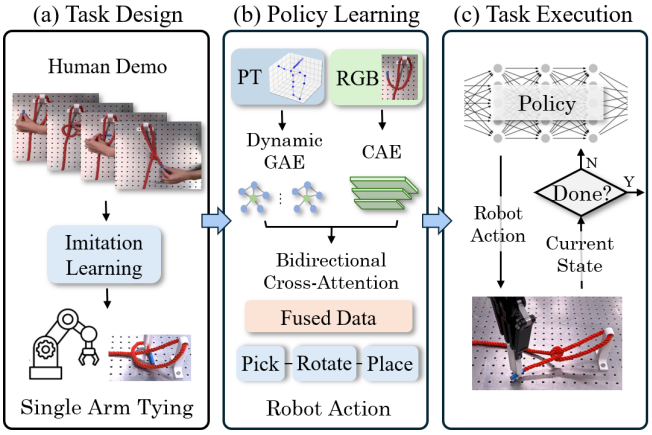


Fig. 1. Overview of our study. (a) Our approach learns to tie hitch knots through human demonstrations. (b) Given an RGB image and 3D keypoints of the rope, the model employs bidirectional cross-attention to fuse multimodal features and predict pick, rotate, and place affordances. (c) The robot executes sequential actions until the task is completed.

As shown in Fig. 1, we propose a framework that learns hitch knot tying policies from visual observations, without relying on perfectly tracked or ordered topological states. Our key insight is that by fusing disordered 3D rope keypoints and RGB scene image, the model can implicitly reason about the rope's physical state and infer effective manipulation actions with limited data, even in the presence of occlusions or crossings. This multimodal bias combines geometric and visual cues, thus constraining the complexity

*The work described in this paper was partially supported by grants AoE/E-601/24-N, 16203223, and C6029-23G from the Research Grants Council of the Hong Kong Special Administrative Region, China.

[1]J. Zuo, B. Zhang, and F. Zhang (corresponding author) are with the Department of Electronic and Computer Engineering, The Hong Kong University of Science and Technology, Hong Kong (email: jzuoai@connect.ust.hk, bzhangcd@connect.ust.hk, eefumin@ust.hk).

of learning and facilitating more efficient model training. Specifically, we employ a Graph Autoencoder (GAE) to encode the topological features from disordered keypoints, while a Convolutional Autoencoder (CAE) processes scene features derived from RGB input. A bidirectional cross-attention mechanism further integrates both modalities to jointly predict pick and place affordances tailored for single-arm robotic knot tying. We collect human demonstration videos to train our policy and evaluate its performance through a series of real-world hitch knot-tying experiments. The results show that our method achieves robust knotting performance across various scenarios, effectively handling challenges related to disordered rope point perception and self-occlusions.

Key contributions of this work include the following:

- We present a rope state representation that extracts disordered 3D keypoints from a potentially occluded rope, suitable for downstream manipulation without fragile tracking of precise topological order.
- We introduce a knotting framework that leverages bidirectional cross-attention to fuse rope geometric features and visual scene features, facilitating the prediction of pick and place affordances for hitch knot tying in visually complex environments.
- The framework is experimentally validated through real-world robotic hitch knot tying with a single arm, demonstrating effective performance under self-occlusion.

II. Related Work

A. State Detection and Tracking of DLOs

Accurate DLO topology estimation for knot tying remains challenging, since vision systems often fail to maintain ordered correspondence under self-occlusion and complex self-crossings. Some works simplify the problem by assuming the rope is fully visible to track during manipulation [13]–[15]. Significant progress has been made in developing visual tracking methods that can handle some occlusions and self-crossings [4], [16], [17]. A common pipeline first extracts an uncrossed initial state via segmentation and refinement [5], [6], subsequently tracks the consistent rope points across frames using non-rigid registration techniques such as Coherent Point Drift (CPD) [2]–[4] or predictive models [18]. However, during knot formation, persistent bending and repeated crossings can cause drift and topology mismatch, leading to incorrect state estimates and downstream manipulation failures [4], [18]. Unlike these tracking approaches, our method foregoes the need for a perfectly ordered topological sequence. Instead, we demonstrate that a robust knot-tying policy can be learned directly from a disordered set of keypoints inputs, which is more reliable in self-occluded scenarios.

B. Robotic Knotting of Ropes

Robotic knot-tying research can be broadly categorized into two main approaches, model-based methods that utilize estimated geometric information [7], [8], [19] and learning-based approaches that employ reinforcement or imitation learning [9]–[11]. Model-based methods typically rely on physically accurate simulations or geometric reasoning based on the estimated state of the rope. Some approaches plan actions by computing the similarity between the current state and a predefined goal state using registration techniques like CPD [8]. Recent study has also explored the application of Reidemeister Moves principles from knot theory [12] to define explainable action primitives for state transitions [7]. A key drawback of these methods is their heavy reliance on precisely tracked state estimation, which is difficult to guarantee in knotting. On the other hand, learning-based methods develop knot-tying policies in a data-driven manner [9], [10], [20]. These strategies can reduce the dependency on precise models and explicit state representations. However, many previous learning approaches are end-to-end, mapping pixel inputs directly to actions, which often results in limited interpretability and a high demand for training data. Our work bridges these two paradigms. We employ a learning-based framework trained on human demonstrations and structure it around an interpretable multimodal state representation that incorporates both disordered rope keypoints and RGB images. This allows the policy to learn actions from human examples effectively with limited data while retaining greater explainability than purely end-to-end methods.

Furthermore, studies on robotic knot-tying can be distinguished by their operational environments: some focus on knotting on a workbench, where tabletop constraints simplify the task, while others address in-air manipulation, which demands more sophisticated handling of unconstrained rope dynamics [9], [21], [22]. In our hitch-tying scenario, the rope is partially supported on a table and partially suspended in air, looped through a hole in a pole, requiring a policy that can reason about both supported and unsupported sections.

III. Problem Formulation

We formulate the task of single-arm hitch knot tying as a sequential decision-making problem with partial observability. The policy π, trained on human demonstration videos of state-action pairs (s_t, a_t), aims to encode the implicit expert dynamic patterns and predict the action

$$a_t = \pi(s_t) \tag{1}$$

which drives the rope toward the desired knotted state. Each action a_t is parameterized as a 3-tuple: (i_t, θ_t, p_t), where $i_t \in \{1, \ldots, N_k\}$ denotes the index of the selected rope keypoint for grasping, $\theta_t \in [-\pi/2, \pi/2]$ specifies the in-plane rotation for gripper orientation during placement, and $p_t \in \mathbb{R}^2$ represents the pixel coordinates for the target placement location.

Since we do not track the fragile topological sequence order of the rope, two major challenges arise:

- Point Set Permutation problem: Due to the lack of consistent correspondence across observations, identical rope states may be represented by different keypoint sequences, as shown in Fig. 2(a). This inconsistency misleads models that rely on fixed keypoint ordering or predefined edge connectivity.

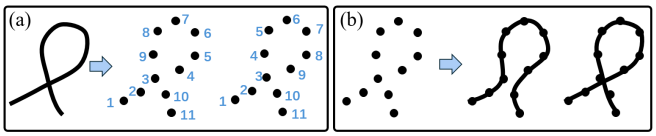


Fig. 2. Challenges in clustering-based rope state estimation without order tracking. (a) *Point Set Permutation*: Identical states can result in different keypoint sequences. (b) *Topological Ambiguity*: Multiple rope configurations may correspond to the same keypoint set.

- Topological Ambiguity problem: Resulting from the discrete keypoint representation, distinct rope configurations can map to identical sets of points, as shown in Fig. 2(b). This ambiguity requires the model to distinguish the underlying topology from disordered points without explicit connectivity.

The first challenge is addressed through our permutation-invariant GAE while the second one is solved with the multimodal fusion mechanism introduced in Section V. Specifically, we represent the rope state using a dynamic graph structure. The multimodal state input of the policy is defined as

$$s_t = f(x^{\text{pt}}, x^{\text{rgb}}) \tag{2}$$

where $x^{\text{pt}} \in \mathbb{R}^{N_k \times 3}$ denotes the untracked and disordered rope keypoints, and $x^{\text{rgb}} \in \mathbb{R}^{H \times W \times 3}$ represents the corresponding RGB image that provides rich contextual information. By integrating both geometric and visual information, the multimodal state s_t helps overcome perceptual limitations and supports the effective prediction of the action a_t.

IV. Rope State Estimation

The rope state estimation module extracts the rope keypoints x^{pt} and their corresponding grasping orientations from the rope point cloud data collected by an RGB-D camera. Accurate rope keypoint registration across frames is challenging due to frequent self-occlusions during the knotting process. Thereby we extract a disordered set of rope keypoints x^{pt} through a clustering approach, avoiding reliance on error-prone tracking of an ordered sequence. The rope point cloud $\mathcal{P} = \{p_j\}_{j=1}^{N}$ is initially segmented into M spatially coherent clusters $\{\mathcal{S}_m\}_{m=1}^{M}$ using Density-Based Spatial Clustering of Applications with Noise (DBSCAN) [23], an unsupervised clustering algorithm which can discover clusters of arbitrary shape and density. For each cluster $\mathcal{S}_m$, we employ K-means clustering to generate K_m representative centroids $\mathcal{C}_m = \{c_{m,k}\}_{k=1}^{K_m}$. The union of these centroids forms our candidate keypoint set, defined as $x^{\text{pt}} = \bigcup_{m=1}^{M} \mathcal{C}_m$.

To determine the local grasping orientation at each keypoint, we utilize Singular Value Decomposition (SVD) rather than predicting it directly, which reduces the model's complexity. For a human-selected (or model-predicted) point p_{pick}, we first identify its nearest neighbor p_c within the point cloud $\mathcal{P}$. A local neighborhood $\mathcal{N}$ is then formed by extracting all points within a radius r of p_c

$$\mathcal{N} = \left\{ p_j \in \mathcal{P} \mid \|p_j - p_c\|_2 < r \right\}. \tag{3}$$

The mean position $\bar{p}$ of these local points is computed as:

$$\bar{p} = \frac{1}{N} \sum_{p_j \in \mathcal{N}} p_j. \tag{4}$$

Then, all points are normalized by the mean value $\bar{p}$ and denoted as

$$\mathbf{X} = [p_1 - \bar{p}, p_2 - \bar{p}, \ldots, p_N - \bar{p}]^{\top}_{N \times 3}. \tag{5}$$

Performing SVD on the centered data matrix $\mathbf{X}$

$$\mathbf{X} = \mathbf{U}\Sigma\mathbf{V}^{\top} \tag{6}$$

where $\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3] \in \mathbb{R}^{3\times3}$, $\Sigma = [\text{diag}(\sigma_1, \sigma_2, \sigma_3), \mathbf{0}] \in \mathbb{R}^{N\times3}$, $\sigma_1 \geq \sigma_2 \geq \sigma_3$. The right singular vector $\mathbf{v_1}$, corresponding to the principal direction of maximum variance in the local point cloud neighborhood $\mathcal{N}$, defines the optimal local grasp orientation. This orientation, aligned with the axial direction of the local rope segment, maximizes the contact area between the gripper and the rope, thereby ensuring a stable grasp.

V. Hitch Knots Tying Framework

The Hitch Knots Tying Framework, illustrated in Fig. 3 and Algorithm 1, consists of three specialized modules: (1) a Multimodal Feature Extraction module employs a dynamic GAE to extract geometry features from x^{pt} and a CAE to extract scene features from x^{rgb}; (2) an Attention-Guided Fusion module that facilitates cross-modal reasoning through bidirectional cross-attention; and (3) Pick and Place Affordance Heads that decode the fused features into action parameters (i_t, θ_t, p_t). This structured approach enables effective policy learning by leveraging both geometric and visual information with limited data.

A. Multimodal Feature Extraction

We self-supervise the extraction of rope keypoint (PT) and RGB image features in the knotting scenarios via reconstruction errors from the GAE and CAE. Given paired samples $\{x^{\text{pt}}, x^{\text{rgb}}\}$, the source features are represented as $\{f^{\text{pt}} \in \mathbb{R}^{n_p \times c^{pt}}, f^{\text{rgb}} \in \mathbb{R}^{h \times w \times c^{rgb}}\}$.

1) Dynamic Graph Autoencoder: The GNN performs message passing between vertices and edges, making it particularly suitable for encoding the topological features of DLOs than general neural network [24]. To address the *Point Set Permutation* problem and achieve consistent rope feature representation, we employ a permutation-equivariant dynamic GAE. The symmetric message passing mechanism guarantees consistent feature encoding regardless of input keypoint ordering. Specifically, we construct a dynamic graph $\mathcal{G}_t = (\mathcal{V}_t, \mathcal{E}_t)$ at each timestep, where vertices $\mathcal{V}_t$ correspond to the keypoints x^{pt} and edges $\mathcal{E}_t$ are dynamically created between vertices within a Euclidean distance threshold. This adaptability allows the graph topology to effectively handle occlusions and deformations. The vertex update rule in our GAE is designed to be permutation-invariant

$$x_i' = \text{AGG}_{j \in \mathcal{N}(i)} \left(\text{MLP}\big(x_i \parallel (x_j - x_i)\big) \right) \tag{7}$$

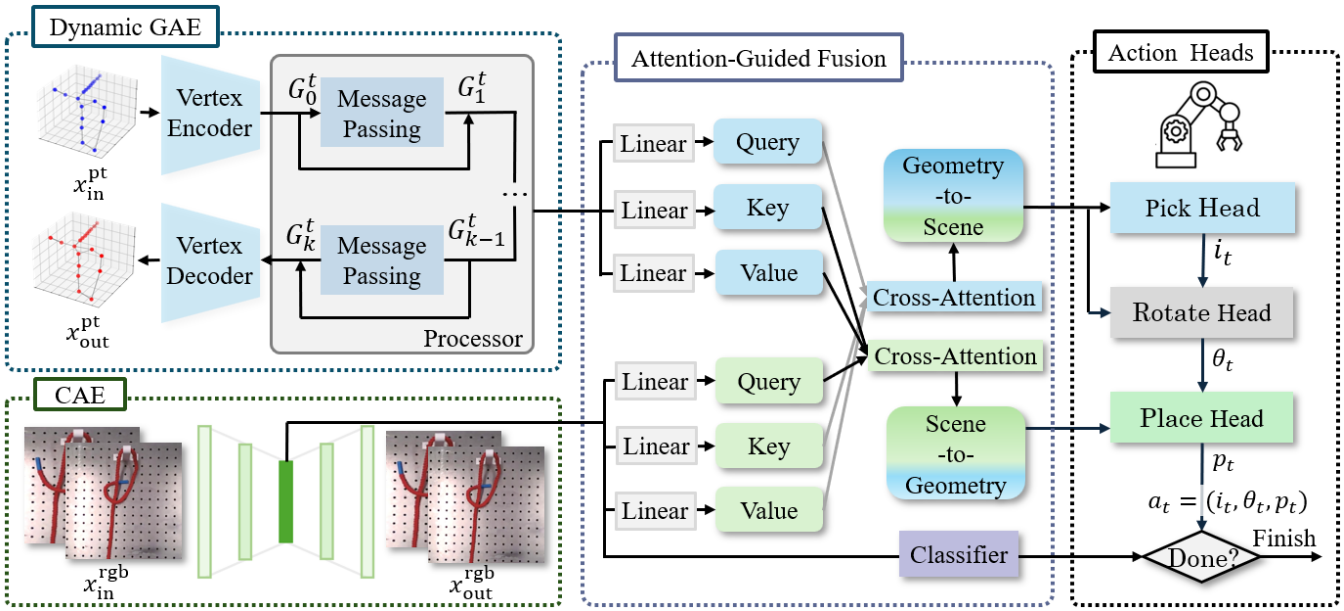


Fig. 3. Overview of the proposed hitch knot tying framework. The system takes rope keypoints and a RGB image as input. A pretrained permutation-equivariant GAE processes keypoints to capture the geometry features, while a pretrained CAE extracts visual features. A bidirectional cross-attention module fuses these multimodal features. Three specialized heads then hierarchically predict the action tuple $a_t = (i_t, \theta_t, p_t)$: the Pick Head selects the grasp point, the Rotation Head predicts in-hand rotation, and the Place Head generates a placement location.

where $\mathcal{N}(i)$ is the set of neighbors of vertex i, x_i denotes current keypoint position, and $x_j - x_i$ represents the relative position vector. The symbol $\|$ denotes concatenation. All edges share identical learnable parameters through a common multilayer perceptron (MLP) and symmetric aggregation functions (AGG), ensuring that the learned features depend solely on the underlying geometry rather than the arbitrary ordering of the keypoints x^{pt}. The GAE is trained with a reconstruction loss to learn a latent encoding f^{pt} that effectively captures the rope's topological state.

2) Image Autoencoder: To extract crucial features from high-dimensional visual information related to multiple actions, we utilize a CAE, which is adept at processing raw images. The CAE consists of an encoder ϕ^I formed by convolutional and fully connected layers, along with a decoder ψ^I composed of deconvolutional and fully connected layers. During training, the parameters are optimized to minimize the reconstruction error

$$\phi^{I*}, \psi^{I*} = \arg\min_{\phi^I, \psi^I} \|x^{\text{rgb}} - (\phi^I \circ \psi^I)(x^{\text{rgb}})\|^2, \tag{8}$$

which allows the network to learn an optimal latent space $f^{\text{rgb}} = \phi^I(x^{\text{rgb}})$ that captures essential information from original image. These image latent features are simultaneously employed to predict task termination through a lightweight MLP-based classifier.

B. Attention-Guided Fusion

While the permutation invariance of our dynamic GAE ensures consistent output features regardless of the observed keypoint ordering, the resulting representation still lacks an inherent physical information along the rope continuum. To address this *Topological Ambiguity*, we integrate PT features with RGB features, which provide complementary visual context regarding the rope's global configuration and spatial relationships.

However, directly concatenating geometric PT features with visual RGB features can lead to spatial-semantic misalignment due to their heterogeneous representations. To enhance the reasoning capabilities from both geometric and visual representations, we employ a bidirectional cross-attention mechanism. This architecture dynamically associates keypoints with relevant image regions, reducing reliance on an explicit ordered keypoint sequence which is often fragile under occlusion or tracking failures. In the geometry-to-scene pathway $f^{\text{pt}\to\text{rgb}}$, queries derived from keypoint features attend to image features f^{rgb} (keys and values), thereby enriching each keypoint with visual context essential for predicting *where to pick*. Conversely, in the scene-to-geometry pathway $f^{\text{rgb}\to\text{pt}}$, queries derived from image features attend to keypoint features f^{pt} (keys and values), incorporating each image pixel with rope geometric information, crucial for predicting *where to place*.

This cross-modal fusion ensures that both geometric and visual information are synergistically integrated, enabling the model to resolve topological ambiguities through complementary cues while maintaining a spatially consistent understanding of the rope's configuration.

C. Action Affordance Head

The action is parameterized as a tuple $a_t = (i_t, \theta_t, p_t)$, where i_t denotes the index of the selected rope keypoint to grasp, p_t specifies the pixel-level placement location, and θ_t determines the rotation on the z-axis from the pick point to the place point. This action space allows for the execution

of complex knotting processes, such as looping, passing, and tightening, using a single-arm robot with a parallel gripper, eliminating the need for handovers or bimanual coordination.

To facilitate learning, each action dimension is discretized. The pick point is chosen from N_k perceived keypoints, the rotation is quantized into N_r bins covering a range from -90° to 90°, and the placement location is discretized at the image pixel level. Instead of directly regressing continuous action coordinates for the grasp and placement points, our approach computes a pick affordance over the rope keypoints and a pixel-wise affordance heatmap for placement. This formulation treats action prediction as a classification problem, which simplifies training [10] and inherently accommodates multimodality in the action space, arising from multiple actions that can transition the rope from the same initial to final configuration.

For action prediction, a naive approach might independently classify each action component. However, the pick-rotate-place operation exhibits strong conditional dependencies. Specifically, the choice of pick point influences feasible rotations, which subsequently constrain valid placement locations. To capture these dependencies without incurring the combinatorial complexity of the joint action space, we factor the action distribution as

$$P(i_t, \theta_t, p_t) = P(i_t)P(\theta_t \mid i_t)P(p_t \mid i_t, \theta_t). \tag{9}$$

This hierarchical decomposition is implemented through specialized network heads. The Pick Head processes the geometry-to-scene features $f^{\text{pt}\to\text{rgb}}$ using a MLP followed by a softmax layer to generate a pick probability distribution across all keypoints. The selected pick point is determined as $i_t = \arg\max_{i_t} P(i_t)$. Subsequently, the Rotation Head utilizes the features of the chosen keypoint alongside $f^{\text{pt}\to\text{rgb}}$ to predict the rotation angle θ_t. Finally, the Place Head combines one-hot encodings of the selected pick point and rotation with the vision-to-geometry features $f^{\text{rgb}\to\text{pt}}$ and processes them through a decoder network to generate a spatial affordance heatmap $\mathcal{H} \in \mathbb{R}^{H\times W}$. The placement location is then selected as the pixel with maximum affordance: $p_t = \arg\max_{p_t} \mathcal{H}(p_t)$.

This cascaded design leverages geometrically enhanced features for picking and visually augmented features for placing. By explicitly modeling action dependencies, our approach simplifies learning and enhances policy stability.

VI. Experiment and Results

A. Hardware Setting

Fig. 5 shows our experimental setup, which consists of a 6-DOF UR 10 manipulator, a wrist-mounted Intel RealSense L515 RGBD camera, and a Robotiq 2F-140 gripper equipped with bio-inspired fingernails [25] to enhance grasping performance. The rope can be initialized in either a crossed or uncrossed shape, and it is distinguished from the background using depth and color filtering. One tip of the rope is marked with blue tape, only to indicate the working end during human manipulation.

Algorithm 1: Algorithm For Hitch Rope Tying

Input:
RGB Image: x^{rgb},
Depth Image: x^{depth},
Task Finish Flag: *done*;
Output:
Robot Action a_t;

Initialization: $done \rightarrow False$;
Initialize $M_{\text{PT}}, M_{\text{RGB}}, M_{\text{CLS}}$ with pretrained weights;
while *not done* **do**
  /* ---- Observation Phase ----- */
  $x^{\text{PT}} = RopeDectction(x^{\text{rgb}}, x^{\text{depth}})$;
  $x^{\text{pt}} = RopeNodeClustering(x^{\text{PT}})$;
  /* ---- Manipulation Phase ---- */
  $f^{\text{pt}}, f^{\text{rgb}} = M_{\text{PT}}(x^{\text{pt}}), M_{\text{RGB}}(x^{\text{rgb}})$;
  $f_{\text{cro}}^{\text{pt}\to\text{rgb}} = M_{FUSE}(f^{\text{pt}}, f^{\text{rgb}})$;
  $f_{\text{cro}}^{\text{rgb}\to\text{pt}} = M_{FUSE}(f^{\text{rgb}}, f^{\text{pt}})$;
  $i_t = PickHead(f_{\text{cro}}^{\text{pt}\to\text{rgb}})$;
  $\overrightarrow{i_t} = SVD(i_t, x^{\text{pt}})$;
  $\theta_t = RotateHead(i_t, f_{\text{cro}}^{\text{pt}\to\text{rgb}})$;
  $p_t = PlaceHead(i_t, \theta_t, f_{\text{cro}}^{\text{rgb}\to\text{pt}})$;
  $a_t = (\overrightarrow{i_t}, \theta_t, p_t)$;
  $RobotCartesianMove(a_t)$;
  $done = M_{\text{CLS}}(f^{\text{rgb}})$;
end

B. Training via Demonstration

We collected 100 human demonstration videos and utilized MediaPipe [26] to extract action points and rotation angles based on the positions and orientations of the thumb and index finger tips, as illustrated in the first row of Fig. 4. From these demonstrations, we derived a set of expert trajectories $\{\tau_1, \tau_2, \ldots\}$, where each trajectory τ consists of a sequence of demonstrated state-action pairs $\{(s_0, a_0), (s_1, a_1), \ldots, (s_n, a_n)\}$, forming our training dataset. In this dataset, the pick affordance of a rope is represented as a one-hot vector, with the element corresponding to the node nearest to the grasping point set to 1. Conversely, the place affordance map is modeled as a two-dimensional Gaussian probability distribution centered at the target placement point. This dataset was then utilized to train the network using a behavior cloning algorithm [27], with the training process conducted on a computer equipped with an NVIDIA GeForce RTX 4080 GPU.

The total loss for the framework is defined as a weighted sum of the individual losses

$$\mathcal{L} = \lambda_{\text{pick}}\mathcal{L}_{\text{pick}} + \lambda_{\text{rot}}\mathcal{L}_{\text{rot}} + \lambda_{\text{place}}\mathcal{L}_{\text{place}}, \tag{10}$$

where $\mathcal{L}_{\text{pick}}$ and $\mathcal{L}_{\text{rot}}$ are cross-entropy losses, $\mathcal{L}_{\text{place}}$ is a negative log-likelihood loss over the Gaussian-smoothed place heatmap, and λ represents the weighting factors for the different losses. The parameter configurations of the proposed method are detailed in Table I.

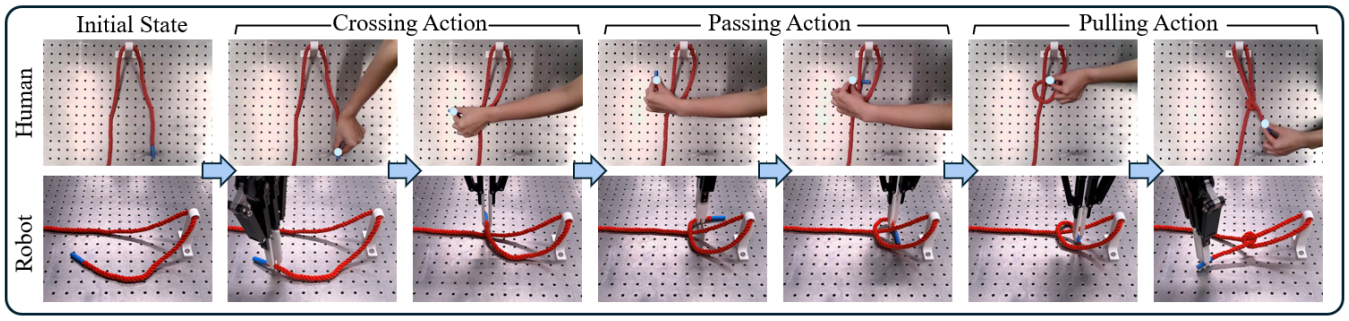


Fig. 4. Examples of sequential motions in hitch knot tying using nylon rope. The human hand motions (top) are derived from the trajectories of the thumb and index finger tips, captured with MediaPipe. The robot (bottom) dynamically infers pick and place actions based on perceptual feedback.

TABLE I
PARAMETER CONFIGURATIONS OF THE PROPOSED APPROACH

GAE Model Parameters	
Encoder	
Hidden Layers	1
Hidden Size	64
Activation Function	ReLU
Processor	
Graph Construction	k-NN (k=3)
Message Passing Steps (k)	2
Hidden Layers	2
Hidden Size	64
Activation Function	ReLU
Decoder	
Hidden Layers	1
Hidden Size	64
Activation Function	ReLU
CAE Model Parameters	
Encoder	
Convolution Layers	3
Channels	$32 \rightarrow 64 \rightarrow 128$
Kernel Size	3×3
Decoder	
Transpose Conv Layers	3
Channels	$64 \rightarrow 32 \rightarrow 3$
Kernel Size	3×3
Classifier & Data Fusion	
Classifier	
Hidden Layers	2
Hidden Size	64
Data Fusion	
Query/Key/Value Dimensions	64
Data & Training Parameters	
Batch Size	8
Image Resolution	640×480
Keypoint Number (N_k)	20
λ_{pick}, λ_{rot}, λ_{place}	2, 1, 2
Learning Rate	5×10^{-4}
Optimizer	Adam
Train/Test Split Ratio	0.8

C. Performance Evaluation

We evaluate our method through a series of real-world experiments, achieving an overall success rate of 84% (42/50 trials) for hitch knot tying with the trained rope. This performance surpasses existing methods as compared in Table II. Notably, our approach outperforms both previous model-based and learning-based techniques in comparable single-arm settings. Fig. 4 depicts sequential examples of the knotting process using a nylon rope, highlighting the robot's capability to dynamically select pick points from disordered keypoints and infer feasible placement locations toward the target configuration.

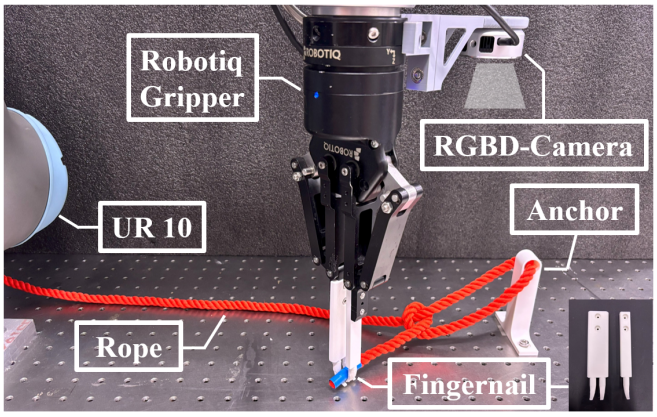


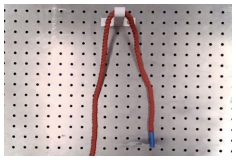
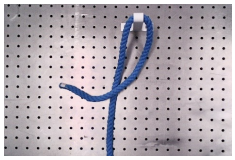
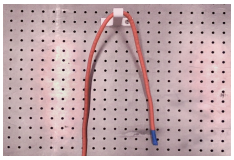
Fig. 5. Hardware setup for conducting rope tying experiments.

To further assess the generalizability of our method, we varied the rope material, diameter, and background settings, as outlined in Table III. The initial state of the rope was always randomized. Our method generalizes well to unseen backgrounds (Scenario 2, 4/5), but performance declines with larger diameter (Scenario 3, 3/5) and fails entirely with polypropylene rope (Scenario 4, 0/5). The failures are primarily attributed to high rope stiffness, resulting from both increased diameter and material differences. Polypropylene exhibits significantly higher stiffness than nylon, leading to strong elastic restoration during bending. This restoration prevents the rope from maintaining the deformed configuration required for knotting, often resulting in task termination. These results indicate a limitation in handling highly stiff materials and emphasize the need for dual-arm coordination.

TABLE II
COMPARISON WITH OTHER KNOTTING METHODS

Reference	Method	Arm	Success Rate
Dinkel *et al.* [7]	Model	Single	50%
Nair *et al.* [10]	Learning	Single	38%
Priya *et al.* [28]	Learning	Single	66%
Pathak *et al.* [29]	Learning	Single	60%
RopeHitch(Ours)	Learning	Single	**84%**

TABLE III
GENERALIZATION EVALUATION WITH OUR KNOTTING METHODS

	Scenario 1	Scenario 2	Scenario 3	Scenario 4
Scenarios				
Diameter	10mm	10mm	14mm	10mm
Material	Nylon	Nylon	Nylon	Polypropylene
Background	seen	unseen	seen	seen
Success Rate	5/5	4/5	3/5	0/5

D. Ablation Study

This section presents ablation studies to evaluate the contribution of individual components within the proposed framework. We examine both PT and RGB modalities independently and compare our bidirectional cross-attention fusion strategy against a direct concatenation baseline.

Fig. 6 shows the training loss curves over 300 epochs. The PT-only modality (case A) achieves a lower training loss compared to the RGB-only modality (case B), indicating that geometric keypoints encode task-relevant features more effectively than visual data alone. Furthermore, our cross-attention fusion approach (case D) achieves a lower final loss than direct concatenation (case C), supporting our hypothesis that naive feature concatenation may lead to spatial-semantic misalignment. Affordance visualization in Fig. 7 reveals complementary characteristics of the modalities: PT-only inputs enable precise grasp affordances but yield ambiguous placement predictions, while RGB-only inputs provide clear placement cues but struggle with accurate grasp localization. Our fused representation (case D) effectively integrates geometric and visual cues, enabling robust predictions for both action steps. As summarized in Table IV, our approach achieves successful outcomes, while other configurations failed completely. This demonstrates the critical importance of effective multimodal fusion.

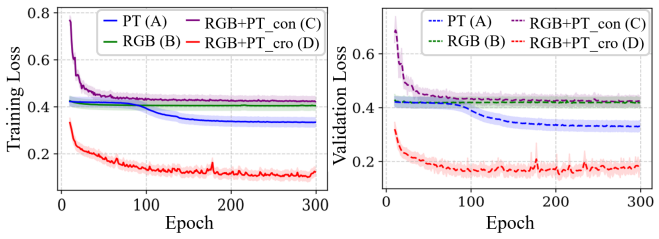


Fig. 6. Training and validation loss curves for different modality configurations. The PT-only input (case A) achieves lower loss than the RGB-only input (case B), while our attention-guided fusion approach (case D) outperforms direct concatenation method (case C).

VII. CONCLUSION

In this paper, we propose a novel framework to construct relatively robust multimodal representations for hitch knot tying. Our approach involves clustering disordered rope keypoints from the rope point cloud and obtaining PT and RGB embeddings through self-supervised learning with a dynamic GAE and a CAE. The bidirectional cross-attention module effectively incorporates implicit interactions between the RGB and PT modalities, enabling these cross-modal embeddings to predict action affordances for manipulating the rope in various states. Generalization experiments have confirmed the effectiveness of our method in the hitch knot tying task, and ablation studies have been conducted to evaluate the contribution of individual components. However, our method is still challenged by low-quality data and the limitations imposed by single-arm manipulation. Future work will focus on developing more robust perception and manipulation methods under complex conditions to enhance our framework's performance.

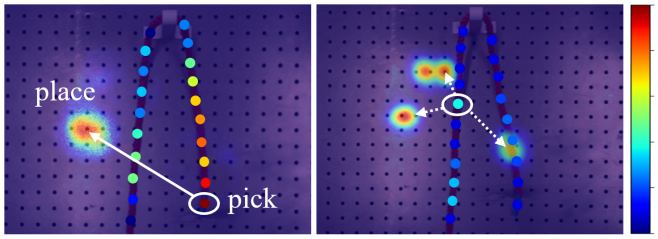


Fig. 7. Comparative analysis of action affordance predictions. (a) Our fused RGB+PT modality accurately predicts kinematically feasible grasp points and well-localized placement regions. (b) In contrast, using RGB-only for grasping and PT-only for placement results in infeasible grasp points and dispersed placement affordances.

TABLE IV
ABLATION STUDY ON DIFFERENT MODULES

Module/Case	Case			
	A	B	C	D
PT modality	✓	×	✓	✓
RGB modality	×	✓	✓	✓
Attention	×	×	×	✓
Success/Fail	F	F	F	S

*PT and RGB indicate input modalities, while Attention denotes cross-attention.

REFERENCES

[1] B. Ai, S. Tian, H. Shi, Y. Wang, T. Pfaff, C. Tan, H. I. Christensen, H. Su, J. Wu, and Y. Li, "A review of learning-based dynamics models for robotic manipulation," *Science Robotics*, vol. 10, no. 106, p. eadt1497, 2025.

[2] A. Myronenko and X. Song, "Point set registration: Coherent point drift," *IEEE transactions on pattern analysis and machine intelligence*, vol. 32, no. 12, pp. 2262–2275, 2010.

[3] T. Tang and M. Tomizuka, "Track deformable objects from point clouds with structure preserved registration," *The International Journal of Robotics Research*, vol. 41, no. 6, pp. 599–614, 2022.

[4] J. Xiang, H. Dinkel, H. Zhao, N. Gao, B. Coltin, T. Smith, and T. Bretl, "Trackdlo: Tracking deformable linear objects under occlusion with motion coherence," *IEEE Robotics and Automation Letters*, vol. 8, no. 10, pp. 6179–6186, 2023.

[5] A. Caporali, K. Galassi, R. Zanella, and G. Palli, "Fastdlo: Fast deformable linear objects instance segmentation," *IEEE robotics and automation letters*, vol. 7, no. 4, pp. 9075–9082, 2022.

[6] A. Keipour, M. Bandari, and S. Schaal, "Deformable one-dimensional object detection for routing and manipulation," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 4329–4336, 2022.

[7] H. Dinkel, R. Navaratna, J. Xiang, B. Coltin, T. Smith, and T. Bretl, "Knotdlo: Toward interpretable knot tying," *arXiv preprint arXiv:2506.22176*, 2025.

[8] T. Tang, C. Wang, and M. Tomizuka, "A framework for manipulating deformable linear objects by coherent point drift," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3426–3433, 2018.

[9] K. Suzuki, M. Kanamura, Y. Suga, H. Mori, and T. Ogata, "In-air knotting of rope using dual-arm robot based on deep learning," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 6724–6731, IEEE, 2021.

[10] A. Nair, D. Chen, P. Agrawal, P. Isola, P. Abbeel, J. Malik, and S. Levine, "Combining self-supervised learning and imitation for vision-based rope manipulation," pp. 2146–2153, 2017.

[11] W. Peng, J. Lv, Y. Zeng, H. Chen, S. Zhao, J. Sun, C. Lu, and L. Shao, "Tiebot: Learning to knot a tie from visual demonstration through a real-to-sim-to-real approach," *arXiv preprint arXiv:2407.03245*, 2024.

[12] V. O. Manturov, *Knot theory*. CRC press, 2018.

[13] M. Yan, G. Li, Y. Zhu, and J. Bohg, "Learning topological motion primitives for knot planning," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 9457–9464, 2020.

[14] M. Yu, H. Zhong, and X. Li, "Shape control of deformable linear objects with offline and online learning of local linear deformation models," in *2022 International Conference on Robotics and Automation (ICRA)*, pp. 1337–1343, 2022.

[15] M. Yu, K. Lv, H. Zhong, S. Song, and X. Li, "Global model learning for large deformation control of elastic deformable linear objects: An efficient and adaptive approach," *IEEE Transactions on Robotics*, vol. 39, no. 1, pp. 417–436, 2023.

[16] A. Caporali, R. Zanella, D. D. Greogrio, and G. Palli, "Ariadne+: Deep learning–based augmented framework for the instance segmentation of wires," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 12, pp. 8607–8617, 2022.

[17] Y. Wang, D. McConachie, and D. Berenson, "Tracking partially-occluded deformable objects while enforcing geometric constraints," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 14199–14205, 2021.

[18] H. Dinkel, M. Büsching, A. Longhini, B. Coltin, T. Smith, D. Kragic, M. Björkman, and T. Bretl, "Dlo-splatting: Tracking deformable linear objects using 3d gaussian splatting," *arXiv preprint arXiv:2505.08644*, 2025.

[19] T. Morita, J. Takamatsu, K. Ogawara, H. Kimura, and K. Ikeuchi, "Knot planning from observation," in *2003 IEEE International Conference on Robotics and Automation (Cat. No.03CH37422)*, vol. 3, pp. 3887–3892 vol.3, 2003.

[20] Y. Yamakawa, A. Namiki, and M. Ishikawa, "Dynamic high-speed knotting of a rope by a manipulator," *International Journal of Advanced Robotic Systems*, vol. 10, no. 10, p. 361, 2013.

[21] S. Kudoh, T. Gomi, R. Katano, T. Tomizawa, and T. Suehiro, "In-air knotting of rope by a dual-arm multi-finger robot," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 6202–6207, IEEE, 2015.

[22] A. Seo, M. Takizawa, S. Kudoh, and T. Suehiro, "Study on tying of a deformable band-shaped object by a dual arm robot," in *2019 IEEE/SICE International Symposium on System Integration (SII)*, pp. 79–84, IEEE, 2019.

[23] E. Schubert, J. Sander, M. Ester, H. P. Kriegel, and X. Xu, "Dbscan revisited, revisited: why and how you should (still) use dbscan," *ACM Transactions on Database Systems (TODS)*, vol. 42, no. 3, pp. 1–21, 2017.

[24] C. Wang, Y. Zhang, X. Zhang, Z. Wu, X. Zhu, S. Jin, T. Tang, and M. Tomizuka, "Offline-online learning of deformation model for cable manipulation with graph neural networks," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 5544–5551, 2022.

[25] J. Zuo, B. Zhang, and F. Zhang, "CaRoBio: 3d cable routing with a bio-inspired gripper fingernail," *arXiv*, 2025.

[26] F. Zhang, V. Bazarevsky, A. Vakunov, A. Tkachenka, G. Sung, C.-L. Chang, and M. Grundmann, "Mediapipe hands: On-device real-time hand tracking," *arXiv preprint arXiv:2006.10214*, 2020.

[27] B. D. Argall, S. Chernova, M. Veloso, and B. Browning, "A survey of robot learning from demonstration," *Robotics and autonomous systems*, vol. 57, no. 5, pp. 469–483, 2009.

[28] P. Sundaresan, J. Grannen, B. Thananjeyan, A. Balakrishna, M. Laskey, K. Stone, J. E. Gonzalez, and K. Goldberg, "Learning rope manipulation policies using dense object descriptors trained on synthetic depth data," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 9411–9418, 2020.

[29] D. Pathak, P. Mahmoudieh, G. Luo, P. Agrawal, D. Chen, F. Shentu, E. Shelhamer, J. Malik, A. A. Efros, and T. Darrell, "Zero-shot visual imitation," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 2131–21313, 2018.