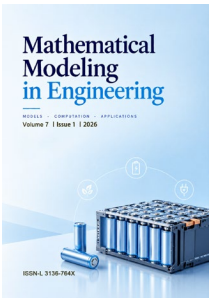
UNIVERSITY OF WEST ATTICA
eISSN: 3136-764X

Mathematical Modeling in Engineering

Vol. 7, No. 2, April, 2026



Cross-Scale SparseFormer-V2 Prediction for Dual-Arm Cable-Routing Operations

Tomislav Matijaš[a,*], Damir Zec[a], Robert Čurić[a]

[a] *Faculty of Electrical Engineering and Computing, University of Zagreb, Zagreb, 10000, Croatia*

Received 08 February 2026; Revised 04 May 2026; Accepted 09 May 2026; Published 15 May 2026

Abstract

Dual-arm cable routing requires a predictor that connects global slack transport with local bending, friction, and fixture contact while operating within a closed-loop control cycle. This paper presents SparseFormer-V2, a cross-scale sparse model that receives synchronized RGB-D observations, cable-centerline samples, dual-gripper states, route waypoints, and candidate contact regions. Cable-centered tokens are organized at fine local, route-contact, and global coordination scales. Mechanically anchored sparse attention links these scales, temporal fusion retains short contact transitions, and shared prediction heads estimate future cable geometry, endpoint error, bend risk, contact state, and near-future routing feasibility over a 1.2 s horizon. Evaluation uses 18,400 simulated sequences and 2,160 real-robot sequences spanning six routing layouts, three cable diameters, two surface materials, and route clearances from 8 to 28 mm. SparseFormer-V2 achieved 92.8% routing-state accuracy, a 5.1 mm endpoint error, a contact F1 score of 0.887, and a success-forecasting F1 score of 0.903 at 18.7 ms per frame. Relative to a dense transformer, it reduced endpoint error by 40.7% and inference latency by 50.7%. Under 45% cable-span occlusion, accuracy remained 86.7%, compared with 77.9% for dense attention. Ablation results attribute the largest gains to cable-centered tokenization and cross-scale interaction. The results support SparseFormer-V2 as a real-time supervisory predictor for constrained cable routing, while self-contact, deep fixture cavities, and branched harnesses remain outside the demonstrated operating range.

Keywords: Dual Arm Cable Routing, SparseFormer V2, Deformable Linear Object, Cross Scale Prediction, Contact Aware Attention

1. Introduction

Cable routing is a common operation in electrical assembly, aircraft harness installation, mobile robot maintenance and service robotics, etc.; however, due to the combined effects of continuous deformation and discrete contact transitions, its automation remains challenging [1]. Geometric center lines are convenient for the planning stage, but they do not show the effects of tension, friction and fixture contact on the actual movement [2]. A globally plausible cable shape may still fail at a local bend, a narrow guide edge, or a short contact patch that is only weakly visible in an RGB-D image [3]. These local effects in dual-arm manipulation are related to the spacing, relative velocity, and grasp transition of the two end-effectors, and an action of one arm may stabilize one segment while disturbing another [4]. Therefore, the problem of implementation is not limited to reconstructing the current curve. The Routing Controller needs to have a short-horizon prediction of endpoint alignment, bend development, contact status and motion feasibility to avoid an incipient error that could lead to an irreversible jam or damage. At the clamps and corners, even a slight deviation within the millimeter range can lead to significant changes in cable tension.

* Corresponding author: tomis.mati@fer.unizg.hr

https://doi.org/10.68406/mme.2026.vol7iss2nm5:54-66

➢ This is an open access article under the CC BY-NC-SA 4.0 license (https://creativecommons.org/licenses/by/4.0/).

© Authors retain all copyrights.

Learning-based perception has improved cable and rope manipulation by using data-driven features instead of hand-crafted ones. Cao et al. [5] combined graph-based cable dynamics with online visual shape servoing for robotic cable assembly, while graph models can represent deformable objects as chains of interacting nodes [6]. Transformers are more effective at learning long-term dependencies and can help address the problem of distant cable segments in tension propagation caused by movements of one gripper [7]. Dense attention is relatively computationally expensive and increases with the growth of image resolution and the length of sampled centerline. The cost of dual-arm routing is relatively high because the predictor needs to reason about visual tokens, robot state, contact candidates and route constraints at control-cycle speed. Sparse Attention is a type of method; however, an arbitrary sparse pattern does not consider the mechanical reality that cable behavior is cross-scale, and both local curvature and global slack affect the feasibility of the next motion.

Therefore, the main research gap is not the lack of another perception backbone. Tang et al. [8] integrated an attention-based deformation predictor with model predictive control and a safety filter, reducing the interface separation between shape estimation, contact reasoning, and action feasibility. Alternatively, a simulation-trained predictor can be used; however, it is unable to show the local failures caused by guide-wall contact and self-shadowing near the grippers [9]. A concise representation of the dual-arm cable task is needed that can be employed in a closed-loop manner, is expressive enough to preserve local bend information, and is suitable for multi-scale signal fusion. The model needs to meet the above requirements, provide some clear intermediate data, and otherwise, it will be difficult for engineers to determine whether the predicted failure is due to endpoint misalignment, excessive bending, or unstable grasping [10].

This study makes three contributions. First, it defines a cable-centered state representation that retains local curvature and contact evidence while encoding route-relative slack and dual-arm coordination at coarser scales. Second, it introduces mechanically anchored sparse attention and compressed temporal fusion so that local failure cues can interact with distant gripper and route states without dense image-wide attention. Third, it evaluates geometry, contact, feasibility, robustness, and runtime within one benchmark containing simulated and real-robot routing sequences. The claimed novelty is therefore the task-specific organization and integration of these components rather than sparse attention in isolation. Section 2 reviews deformable-object perception and predictive models. Section 3 formulates the routing state and cross-scale tokenization. Section 4 describes SparseFormer-V2 and its training objective. Section 5 presents comparative, ablation, robustness, and failure analyses. Section 6 states the conclusions, limitations, and future research directions.

2. Related Work

2.1. Perception and State Representation for Deformable Linear Objects

Perception of deformable linear objects has generally started with shape extraction. Traditionally, pipelines have employed edge detection, skeletonization and geometric smoothing to find the centerline of cables; while effective in well-lit conditions, they are prone to self-intersection or contact with fixtures [11]. Later, RGB-D sensing enabled depth-aware centerline recovery and improved segmentation of background clutter. These methods are still appealing for industrial cells because they are interpretable and can be trained with limited data; however, their output often only provides the current status. Routing tasks need a short forecast of how the cable will move after both arms apply coordinated displacement, and static centerline extraction does not record the tension and bending history that cause this motion.

Graphs and particles have been used to connect perception with physical reasoning. A cable can be considered an ordered chain of nodes, and both axial distances and local curvatures are taken to be edges [12]. These models can be connected to a controller easily, and the motion of the end-effector, local strain, and collision risk can all be displayed on the same graph. In fact, to draw a good graph, the nodes must be placed precisely; however, self-occlusion may occur, and some of the cables will be covered. Hybrid models integrate neural segmentation with physically-motivated graph updates; however, they often require hand-tuned contact logic and are not easily generalizable to different cable thicknesses or surface frictions.

Dual-arm manipulation introduces coordination variables not present in single-arm datasets. The two grippers can change the tension of the cable in opposite directions, and the route will fail if one arm retracts and does not leave enough slack for local insertion of the other arm. Previous research on bimanual deformable manipulation has focused on coordinated grasping, tension regulation and fixture-aware planning [13]. The studies have shown the need for robot state, but many perception models still process the scene image without an explicit dual-arm state channel. A cable predictor that does not consider gripper velocity, arm spacing and planned waypoint relationships cannot determine reliably whether a curve is feasible or if it will become infeasible after the next motion.

Another problem is current contact reasoning. Contact with the guide, clamp or tray edge will limit the range of motion by reducing the space available, and thus lead to a failure due to overbending. Some contact-aware methods predict the probability of collision or infer support relationships from depth discontinuities [14]. They have the

advantage of rich local physical details, but they rarely include contact information with global slack data in a single learned representation. This division is not convenient for cable routing, and the same local contact status may be safe in one part of the network but unsafe at the global endpoint.

2.2. Sparse Attention and Predictive Modeling for Robotic Manipulation

Transformer models have been applied to robotic perception to address the problem of long-range spatial and temporal relationships. Dense self-attention can connect far-apart sections of the cables, gripper data and route nodes without explicit neighborhood specification [15]. Cable routing is also relatively flexible; a small displacement of the gripper can cause a distant section of the cable to stretch after the transmission of tension. The problem is a quadratic cost of dense token interactions. High-resolution visual features, temporal windows and robot-state tokens lead to very slow dense attention for industrial control loops that need to achieve prediction frequencies of tens of Hz.

Sparse and deformable attention reduce computation by selecting a small number of relevant tokens. Deformable attention in vision tasks has shown that a model can learn where to sample rather than comparing every token with every other token [16]. Given that the main parts of cable manipulation are located near the centerline of the cable, grippers, contact candidates and route waypoints, this idea can be applied. Direct application of general sparse vision models is not feasible; therefore, the sampling pattern must be based on cable mechanics rather than image saliency. Add fine-curvature details near the probable contact point and use a coarser representation for the rest of the slack.

Recurrent networks, temporal convolution, and sequence transformers have been employed for temporal prediction. Recurrent models are relatively small and have been applied to predict robot states; however, they may smooth out short-duration contact phenomena of only a few frames [17]. Sequence Transformers can keep more temporal information, but they are computationally more expensive for a large number of spatial tokens per frame. The two attributes required by the predictor for cable routing are: memory of the current shape of the cable and a rapid response to a new contact or bend event. Thus, a model for sparse temporal fusion can be constructed after the spatial tokens have been organized according to cable geometry.

Recent robot-learning systems increasingly combine perception, action context, and feasibility prediction within a shared network, which can reduce interface losses and propagate uncertainty across tasks [18]. Existing approaches nevertheless tend to emphasize either image-level representation, generic action prediction, or high-level collaboration. They do not directly organize the representation around the mechanical scales of cable routing: local curvature and contact, route-relative slack, and dual-arm coordination. SparseFormer-V2 addresses this gap through cable-centered tokens, cross-scale sparse interactions, and joint heads for shape, contact, and routing feasibility.

3. Problem Formulation and Cross Scale Representation

3.1. Dual Arm Routing State and Prediction Targets

The task considered here is dual-arm routing of a flexible cable through known guide waypoints, clamps, and insertion regions. The RGB-D camera is assumed to cover the active routing workspace, the cable ends remain fixed in the two grippers during each prediction window, and route geometry is available in the camera-aligned workspace. Motions are sufficiently slow for inertial effects to be secondary over the 1.2 s forecast horizon, although frictional contact and evolving curvature are retained. At each time step, the input comprises the current RGB-D observation, the poses and velocities of both grippers, the planned route centerline, and fixture geometry. The output is a short-horizon prediction of cable shape, endpoint alignment, bend risk, contact state, and motion feasibility; it is not a long-horizon motion planner or a force controller.

A cable instance is represented as an ordered set of centerline samples and a smaller set of routes and contact tokens. Let the visible cable samples be denoted by x_i, the left and right gripper states by g_L and g_R, and the route waypoint set by q_j. SparseFormer-V2 does not treat every image patch as equally important. Instead, it forms cable-centered tokens by sampling features around the estimated centerline, guide boundaries, and gripper neighborhoods. The sampling bands are wider near occlusion and expected contact regions, where visual uncertainty is larger. This construction gives the network access to local geometry without forcing dense attention over the entire image.

For a compact description of the observation available at time t, the dual-arm routing state is expressed by the equation below.

$$S_t = \{X_t, G_t, Q, C_t\}, X_t = \{x_i = (u_i, v_i, z_i, s_i)\}_{i=1}^{N} \quad (1)$$

Within this equation, X_t denotes ordered cable samples, G_t stores the left and right gripper states, Q represents route waypoints and fixture constraints, and C_t stores candidate contact regions. The predictor receives a recent observation window and estimates the state at a short future horizon.

3.2. Cross Scale Cable Tokenization

The first scale captures local cable mechanics. Each token stores image features, depth residuals, normalized arc length, curvature proxy, and distance to the nearest route boundary. Local curvature is treated as a predictive signal rather than a post-processing quantity because bend risk often appears before a complete routing failure. The model therefore learns whether a small curvature change is safe, caused by useful contact, or likely to become a high-friction bend. This is important in narrow routing channels where clearance is limited and a visually minor bend can prevent the cable from entering a clamp.

The second index is overall slack and dual-arm coordination. Slack is not a single number, otherwise, the same amount of slack could be added to all parts of the path or concentrated near the collision corner. SparseFormer-V2 uses route-relative features to encode the location of slack, changes in arm spacing, and alignment between the planned motion direction and the cable tangent. Figure 1 is used to show the resulting problem formulation and token construction flow. Display sensor inputs, centerline and route marking, dual-arm state fusion, and predicted outputs used by the controller.

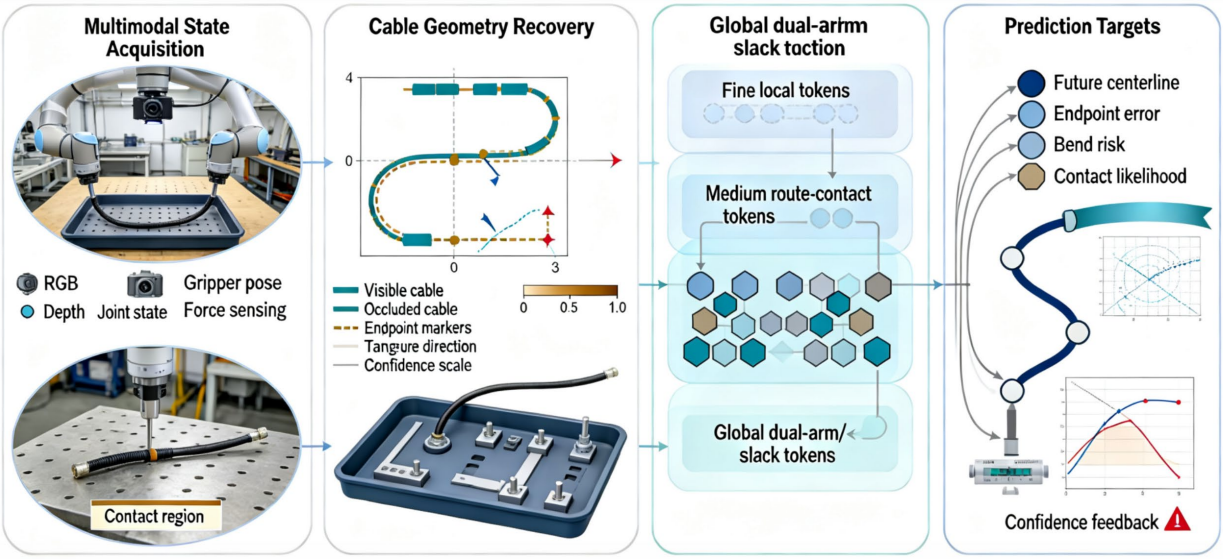


Figure 1. Problem formulation and cross-scale token construction workflow

Token construction also carries an explicit visibility state so that missing depth is not confused with a valid background observation. For each centerline sample, a confidence value is estimated from color segmentation, local depth continuity, and agreement with the preceding frame. Low-confidence samples remain in the ordered sequence but receive a broader candidate neighborhood, allowing route and gripper tokens to contribute more strongly. Route coordinates are normalized by the active fixture span, while distances and curvature are scaled by cable diameter. This normalization lets one representation cover the 4, 6, and 8-mm cables without making a fixed pixel radius carry different physical meanings. Contact candidates are generated at route boundaries, detected depth discontinuities, and points where the centerline approaches a fixture within one cable diameter. A binary mask distinguishes observed from inferred candidates, enabling the prediction head to express uncertainty when the cable passes behind a clamp. The resulting token set preserves arc-length order, physical scale, and observation quality while remaining compact enough for repeated evaluation inside a control loop.

The feature channels and scale identifiers are combined through the following embedding equation.

$$e_i = \phi\left(\left[f_i, \kappa_i, d_i^r, d_i^g, o_i\right]\right) + \psi(s_i) + \chi(m_i) \tag{2}$$

In the embedding equation, f_i is the local visual-depth feature, κ_i is the curvature proxy, d_i^r and d_i^g describe route and gripper distances, o_i denotes occlusion confidence, s_i is normalized arc length, and m_i marks the token scale. This construction allows the same attention block to process local and global information without erasing scale identity.

4. SparseFormer V2 Prediction Model

4.1. Sparse Attention Backbone

The four modules of SparseFormer-V2 are: cable-centered tokenization, cross-scale sparse attention, temporal fusion, and task-specific prediction heads. The three sets of tokens generated in the tokenization stage are: fine local tokens for high-curvature or contact-prone areas, medium route tokens near fixtures and waypoints, and global coordination tokens

indicating gripper status and slack distribution. These groups have the same embedding dimension but keep scale labels, and thus the attention layer can learn different sampling budgets for local and global information. The number of attended interactions will be approximately proportional to the number of cable tokens in the architecture.

Sparse attention is computed by allowing each query token to sample a bounded set of offsets around mechanically defined anchors. Fine-scale anchors are placed at high-curvature samples, depth discontinuities, and route-boundary proximities. Medium-scale anchors represent fixtures and waypoints, whereas global anchors represent both grippers and slack-bearing cable sections. The offset predictor may refine these anchor locations, but it cannot introduce unrestricted image-wide connections. With N tokens and K sampled locations per query, aggregation requires O(NK) interactions, where K is fixed, rather than the $O(N^2)$ interactions of dense self-attention. This bounded sampling rule makes inference time predictable and keeps the sparse pattern tied to cable geometry rather than generic visual saliency.

Sparse aggregation for each query token is governed by the equation below, with normalized weights over a bounded set of learned sampling locations.

$$A_i = \sum_{k=1}^{K} \alpha_{ik} W_v e(p_i + \Delta p_{ik}), \sum_{k=1}^{K} \alpha_{ik} = 1 \tag{3}$$

As specified in this equation, the learned offsets Δp_{ik} select mechanically relevant tokens around each anchor p_i, and the attention weights α_{ik} determine which sampled features dominate the update. The limited sampling budget K keeps inference predictable while still admitting nonlocal interactions.

4.2. Temporal Fusion and Multi Head Prediction

Temporal fusion uses a sliding window of recent token states. Rather than feeding full image features from all frames into a sequence transformer, SparseFormer-V2 first compresses each frame into cable-centered tokens and then performs sparse temporal attention over the compressed sequence. This ordering reduces latency and limits the effect of irrelevant background motion. The temporal block retains short contact events by assigning separate memory tokens to bend risk, endpoint alignment, and grasp exchange quality.

Temporal fusion over the recent observation window is summarized by this state-update equation.

$$H_t = T_s(E_{t-L+1}, \dots, E_t, G_{t-L+1:t}, Q) \tag{4}$$

The equation uses a sparse temporal operator T_s in which each memory token attends only to scale-compatible tokens and a small number of cross-scale anchors. This preserves frame-to-frame contact information while avoiding dense sequence attention over the full visual field.

The prediction heads share the fused representation but specialize the final mapping. The geometric head predicts future centerline displacement and endpoint error. The contact head predicts fixture contact and bend risk. The feasibility head predicts whether the next planned dual-arm motion should proceed, slow down, or be replanned. The heads are trained jointly because the failure modes are coupled: excessive bend, poor endpoint alignment, and unstable grasp transition can appear in the same short horizon. Figure 2 is reserved for the complete SparseFormer-V2 architecture, including token groups, sparse attention, temporal fusion, and multi-head prediction outputs.

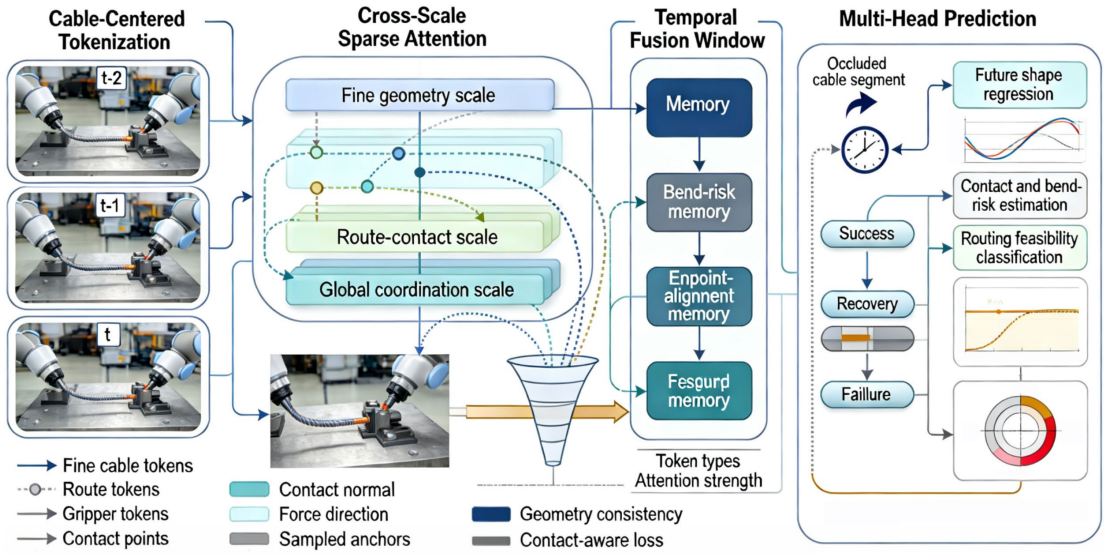


Figure 2. SparseFormer-V2 architecture with cross-scale sparse attention, temporal fusion, and multi-head prediction

Training uses complete routing clips rather than independently shuffled frames. Each mini-batch contains a contiguous observation window, and the same geometric transformation is applied to every frame in that window. Color, depth noise, friction, and cable stiffness are perturbed without changing route topology or temporal order. The four prediction heads are optimized jointly from initialization. Geometric and bend-risk losses are normalized by their training-set scales, while class-balanced weights are applied to contact and feasibility labels. Loss weights and the stopping epoch are selected only on the validation split, with endpoint precision and early failure recall treated as co-primary criteria. Test sequences are not used for threshold selection or model selection.

Joint optimization of the prediction heads is defined by the final training equation.

$$\mathcal{L} = \lambda_x \left\| \hat{X}_{t+\tau} - X_{t+\tau} \right\|_1 + \lambda_b |\hat{b} - b| + \lambda_c BCE(\hat{c}, c) + \lambda_y CE(\hat{y}, y) \tag{5}$$

The training equation combines future shape regression, bend-risk regression, contact classification, and routing feasibility classification. Its weights were selected on the validation set to balance endpoint precision with early failure prediction.

5. Experimental Data and Model Analysis

5.1. Experimental Setup and Overall Comparison

The benchmark contains six routing-layout families: straight guide, single corner, double corner, clamp insertion, crossing avoidance, and constrained tray routing. It includes 18,400 simulated sequences and 2,160 real-robot sequences covering three cable diameters, two surface materials, and route clearances from 8 to 28 mm. Training uses the simulated set and 70% of the real sequences. Validation and test partitions are formed at the complete-sequence level, so adjacent frames from one routing episode cannot occur in different partitions; held-out layouts and lighting conditions are retained for evaluation. Synthetic-to-real adaptation is restricted to color, depth-noise, stiffness, and friction randomization because excessive scene-specific augmentation can mask genuine contact errors in deformable-object evaluation [19]. Each sequence provides synchronized RGB-D frames, arm joint states, gripper poses, centerline labels, route waypoints, contact annotations, and a binary record of whether the next 1.2 s motion segment was completed without replanning or severe contact.

The five baseline predictors to be selected for comparison are a CNN-LSTM predictor, a graph neural network with centerline nodes, a temporal convolutional network, a dense transformer, and a deformable attention model without cross-scale cable tokenization. The optimization of the hyper-parameters used the same validation set, and all the reported values are the mean of five training seeds. Most closed-loop industrial perception systems have a short control-cycle latency and do not require high accuracy because of the reduced computation [20]. Thus, latency and memory were combined with prediction accuracy for evaluation. Endpoint error was calculated in the camera-aligned workspace after hand-eye transformation, and bend risk was normalized by curvature energy and contact proximity. A sample was marked as routing-feasible only if all the thresholds for endpoint error, contact state and bend risk were exceeded; otherwise, a visually plausible but mechanically invalid shape did not receive a positive label.

Table 1 shows that SparseFormer-V2 reached 92.8% routing-state accuracy, exceeding the dense transformer by 4.4 percentage points and the deformable-attention model by 5.9 points. Its 5.1 mm endpoint error was 40.7% lower than the dense-transformer error of 8.6 mm, and its bend-risk mean absolute error decreased from 0.103 to 0.067. Contact F1 and success F1 improved simultaneously, indicating that the geometric gains were not obtained by relaxing the binary decision criteria. The reported entries are means across five training seeds; because seed-level confidence intervals are not provided, the comparisons are interpreted as descriptive rather than inferential. Benchmark evidence indicates that tail endpoint errors and contact-state mistakes are more consequential than small mean centerline deviations [21]. Consistent with that observation, the 90th-percentile endpoint error was 8.9 mm for SparseFormer-V2 and 14.8 mm for the dense transformer. The remaining tail errors occurred mainly during fixture insertion, where a local offset can leave most of the visible centerline close to its label while preventing the cable from reaching the fixture throat.

Table 1. Quantitative comparison on the dual arm cable routing benchmark

Model	Accuracy (%)	Endpoint Error (*mm*)	Bend MAE	Contact F1	Success F1	Latency (*ms*)
CNN-LSTM	81.6	12.4	0.142	0.761	0.782	13.5
Graph Predictor	84.7	10.2	0.128	0.803	0.817	21.8
Temporal ConvNet	83.9	11.1	0.134	0.792	0.806	15.6
Dense Transformer	88.4	8.6	0.103	0.842	0.841	37.9
Deformable Attention	86.9	9.1	0.111	0.831	0.852	22.4

SparseFormer-V2	92.8	5.1	0.067	0.887	0.903	18.7

Figure 3(a) separates accuracy by route geometry. SparseFormer-V2 ranged from 95.1% on straight guides to 90.4% on constrained trays, whereas the dense transformer dropped to 84.2% on trays; the widening margin indicates that route-contact anchors become more valuable as local constraints accumulate. Figure 3(b) shows endpoint error for 4, 6, and 8-mm cables. The proposed model increased only from 4.6 to 5.7 mm across this range, compared with 7.8 to 9.4 mm for dense attention, suggesting that scale-coded tokens limited sensitivity to apparent cable width. Figure 3(c) reports success forecasting under short, medium, and long slack. Its advantage over the dense transformer grew from 0.050 to 0.071 F1 as slack increased, which is consistent with global tokens carrying information that a local segment cannot expose. Figure 3(d) compares predicted and measured bend risk. A fitted slope of 0.94 and an R-squared value of 0.89 show close agreement over the risk range, while the denser scatter of baseline errors at high risk explains why several geometrically acceptable predictions were still assigned an unsafe feasibility state.

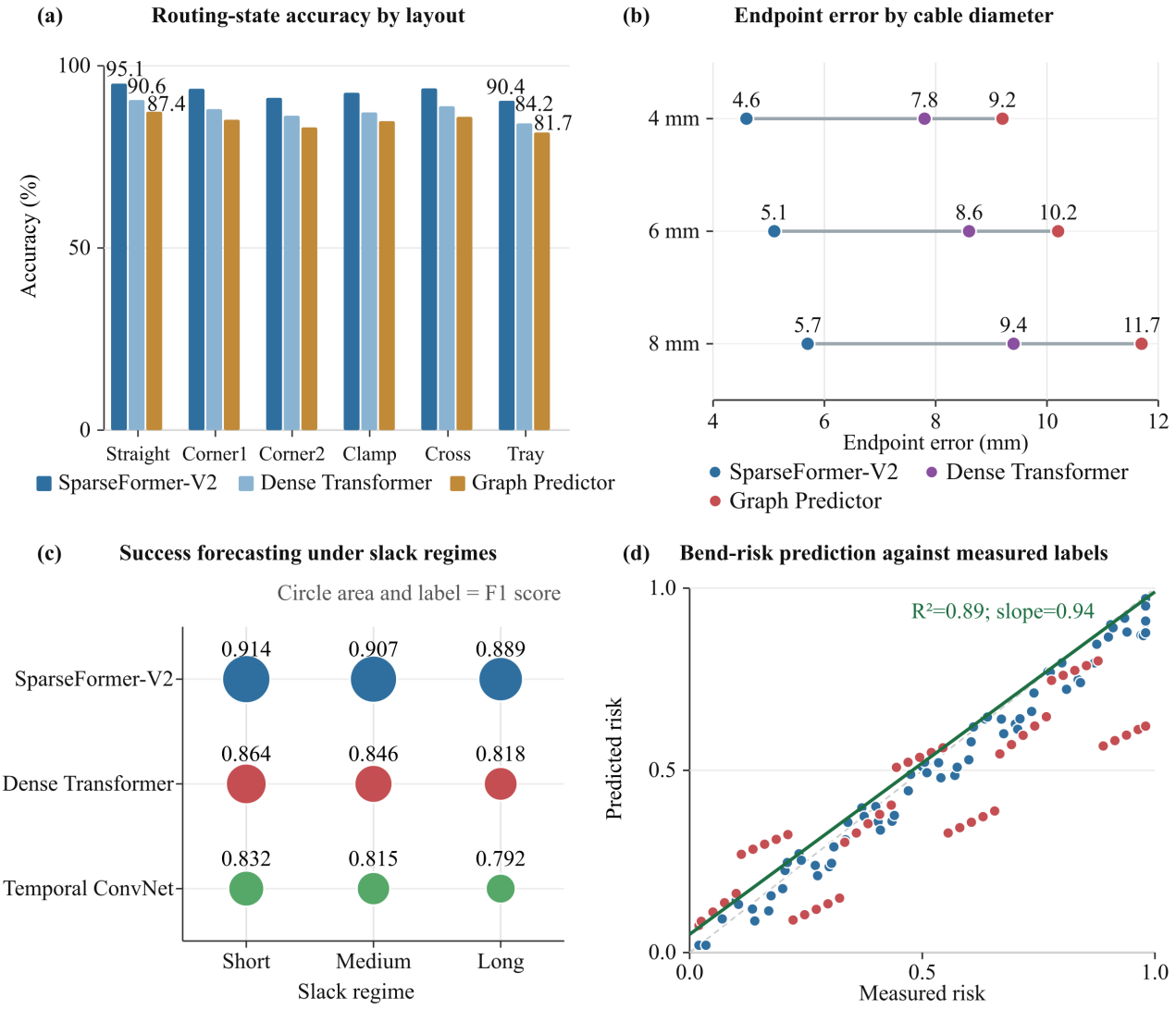


Figure 3. Comparative prediction performance across routing conditions: (a) routing-state accuracy by layout; (b) endpoint error by cable diameter; (c) success forecasting under slack regimes; (d) bend-risk prediction against measured labels

5.2. Ablation and Efficiency Analysis

Contact-supervised deformable-object models have shown that whether physical feasibility is mentioned explicitly or inferred from appearance [22] affects the results. Table 2 shows the same mode in the benchmark. Removal of cross-scale attention reduced accuracy from 92.8% to 89.1% and increased bend-risk error from 0.067 to 0.094. Removal of temporal fusion resulted in a smaller geometric penalty, but the success F1 dropped to 0.861 because a single frame could not distinguish between stable contact and evolving slip. Uniform image tokens resulted in the largest accuracy drop and an increased latency of 29.6ms; thus, indiscriminate visual sampling wasted computation without retaining the route-relative structure required by the prediction heads. The no-contact-loss variant achieved a reasonable end-point accuracy but increased the bend-risk error to 0.093 and thus did not show the advantage of general feature regulation.

Table 2. Ablation analysis of SparseFormer-V2 components

Variant	Accuracy (%)	Endpoint Error (*mm*)	Bend MAE	Success F1	Latency (*ms*)
Full SparseFormer-V2	92.8	5.1	0.067	0.903	18.7
No Cross-Scale Attention	89.1	6.8	0.094	0.874	16.9
No Temporal Fusion	90.2	5.9	0.083	0.861	17.4
Uniform Image Tokens	87.6	7.5	0.101	0.848	29.6
No Contact-Aware Loss	90.5	6.2	0.093	0.879	18.1

Figure 4(a) quantifies the accuracy contribution of each component: cross-scale attention added 3.7 percentage points, and cable-centered tokenization added 5.2 points relative to uniform tokens. The larger tokenization effect shows that attention sparsity is useful only when candidate tokens already encode cable and route relevance. Figure 4(b) jointly displays latency and memory. SparseFormer-V2 used 2.9 GB at 18.7 ms, while dense attention required 5.8 GB at 37.9 ms; the approximately twofold compute gap is larger than the accuracy gap and matters for control-cycle regularity. Figure 4(c) varies the sparse sampling anchors from 4 to 20. Contact F1 rose from 0.821 to 0.887 at 12 anchors, then declined to 0.876 at 20, indicating that excessive anchors admitted visually salient but mechanically irrelevant background regions. Figure 4(d) gives feasibility calibration error. SparseFormer-V2 achieved 0.041, compared with 0.076 for the dense transformer; reliable probability calibration is required when a replanning rule acts on confidence rather than on a hard class alone [23].

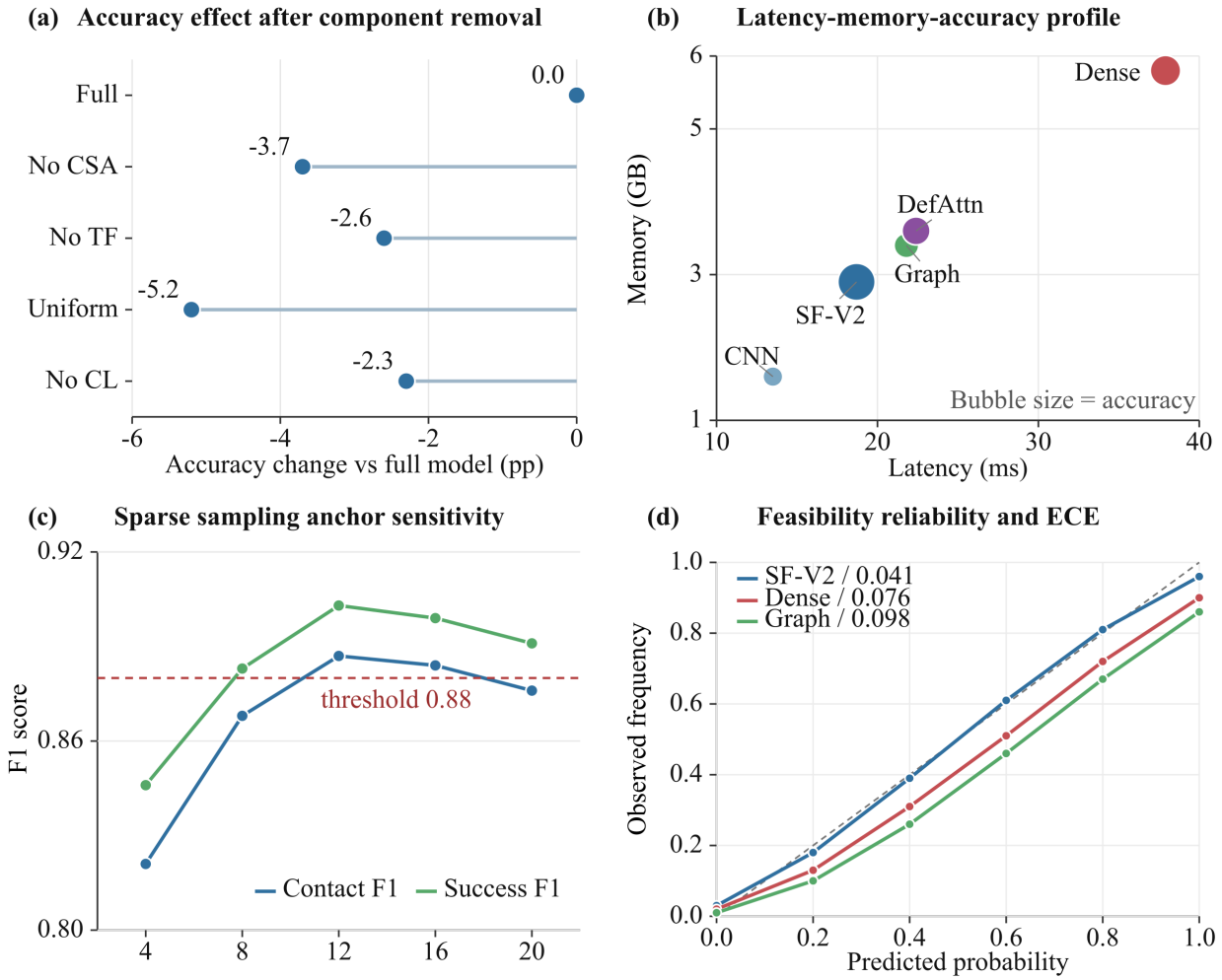


Figure 4. Ablation, efficiency, and calibration behavior: (a) accuracy loss after component removal; (b) latency and memory profile; (c) sparse sampling anchor sensitivity; (d) feasibility calibration error

Optimizations have been made at the level of tokens to improve efficiency without reducing wall-clock time. SparseFormer-V2 used 1,024 cable-centered tokens and 96 route-contact tokens per frame, had an average attention density of 8.7%, and reduced multiply-accumulate operations by about 46% compared to a dense transformer. The standard deviation of the latency in the test sequence was 2.1 ms, and no sequence exceeded 27.8 ms for more than four

consecutive frames. It is more suitable for operation than that of a model with a similar mean and periodic long stalls because the dual-arm controller conducts perception updates at regular intervals. The whole model took 6.4 hours to run 120 training rounds on a single industrial GPU, and the uniformization of tokens took 8.9 hours due to the extra forward computation and gradient storage for the expanded attention set.

The ablation results separate geometric prediction from decision accuracy. Removing temporal fusion increased endpoint error by 0.8 mm and reduced success F1 by 0.042, suggesting that temporal evidence contributes more strongly to the direction of contact evolution than to instantaneous centerline fitting. Removing cross-scale attention affected both geometric and decision metrics, whereas uniform image tokens produced the largest accuracy loss and a substantial latency increase. These patterns indicate that efficiency is not produced by sparsity alone: the candidate tokens must first preserve cable, fixture, and route relevance. The ablation evidence supports the chosen architecture, although the reported means should not be interpreted as statistical significance without seed-level intervals.

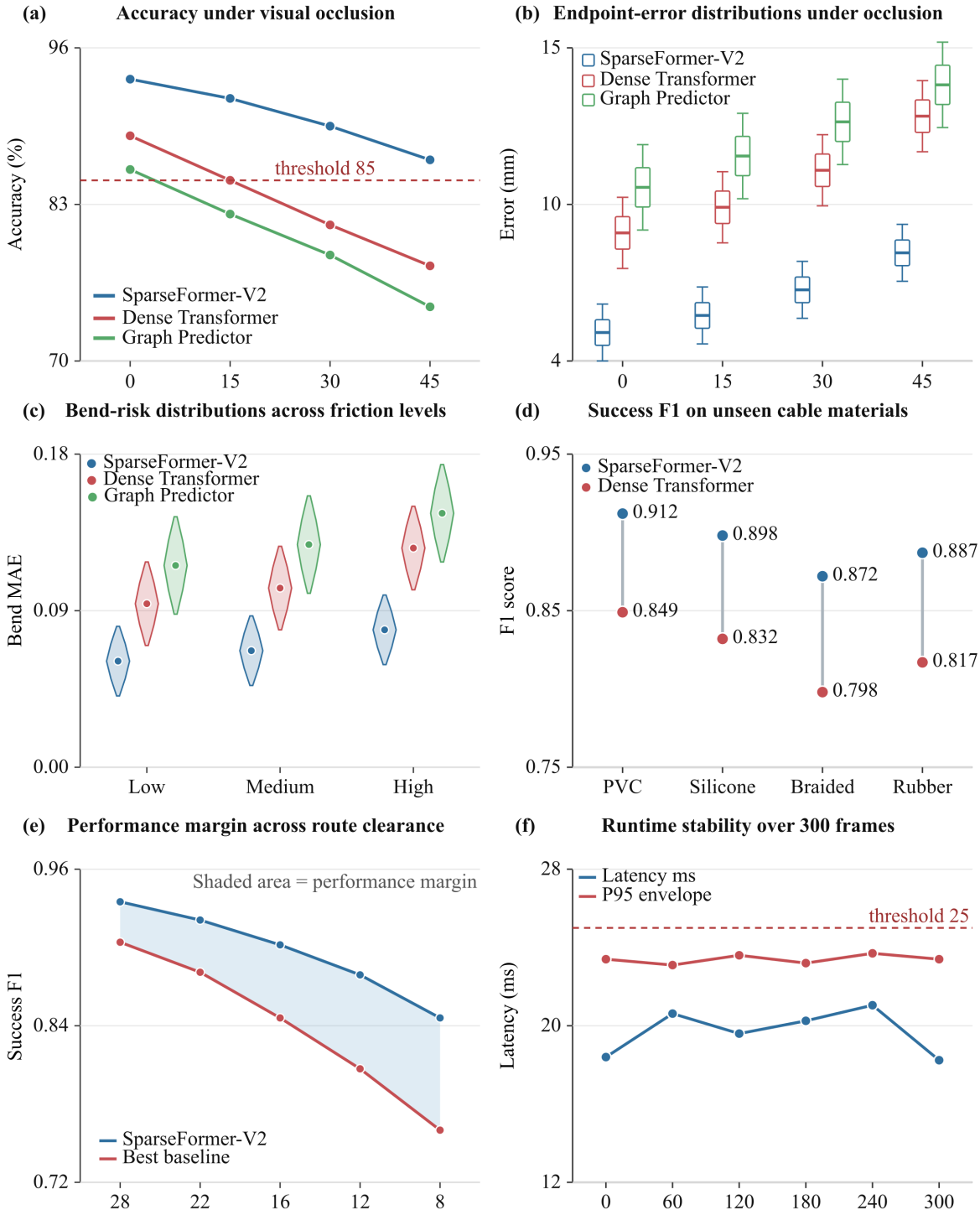


Figure 5. Robustness tests under occlusion, friction, material shift, and runtime stress: (a) accuracy under visual occlusion; (b) endpoint error under visual occlusion; (c) bend-risk error across friction levels; (d) success F1 on unseen cable materials; (e) sensitivity to route clearance; (f) runtime stability over 300 frames

5.3. Robustness and Failure Pattern Analysis

Robustness to incomplete vision and material shift remains a persistent challenge in deformable-object perception. Figure 5(a) shows that SparseFormer-V2 accuracy decreased from 93.4% to 86.7% as cable-span occlusion rose from 0% to 45%, while dense-transformer accuracy fell from 88.7% to 77.9%; the expanding margin indicates that arm and route tokens compensated for missing centerline evidence. Figure 5(b) shows the corresponding endpoint errors of 5.0 to 7.8 mm for the proposed model and 8.5 to 12.6 mm for dense attention. Figure 5(c) reports bend-risk error across low, medium, and high friction, where SparseFormer-V2 remained between 0.061 and 0.079 rather than following the baseline rise to 0.126. Figure 5(d) evaluates unseen materials; braided insulation was hardest, but the proposed model retained 0.872 success F1. Figure 5(e) shows that its margin over the strongest baseline expanded from 0.031 at 28 mm clearance to 0.086 at 8 mm, locating the largest benefit in mechanically constrained routes. Figure 5(f) plots a 300-frame sequence in which the 95th-percentile latency remained 23.4 ms and the brief peaks did not persist across control cycles. This widening degradation under missing visual evidence is consistent with the robustness limitations reported for deformable-object perception benchmarks [24].

Contact transitions are a recognized source of deformable-manipulation failure even when steady-state shape error remains small [25]. Among the 312 failed test sequences, mutually exclusive labels assigned during replay review attributed 39.7% to underestimated fixture contact, 27.6% to excessive slack near the second gripper, 19.2% to occluded crossings, and 13.5% to route labels that were geometrically feasible but physically unstable. Depth ambiguity is also known to limit state estimation when overlapping cable layers have nearly identical ranges [26]. This limitation was evident at crossings: when more than one-third of the centerline was simultaneously occluded, route tokens constrained the possible path but did not reliably recover layer order. Attention inspection was used only as a diagnostic aid. Contact-token mass increased from 22% during free-space transport to 39% during clamp insertion, while underestimated-contact failures showed weak activation near the fixture edge. These observations are consistent with missing local evidence, but they do not by themselves establish a causal explanation for the prediction error.

The Arm Spacing function is an indirect tension-limiting device for the current routing problem. Figure 6 is the interaction of the six arm spacings and four curvature levels. Success reached 96.2% at 260 mm with a low curvature, but dropped to 71.4% at 140 mm because of very high curvature. For a medium-sized curve, the 220-300mm band was still more than 90% full; both narrow and wide spacing resulted in failure: the narrow spacing formed a compact loop, and the wide spacing had used up all the slack before insertion. Earlier bimanual manipulation studies have also shown that gripper separation can be used to adjust cable tension [27]. In the low-spacing, high-curvature cells, the model increased the bend risk before the endpoint error exceeded the threshold and enabled replay-based replanning in 83.5% of these cases. Asymmetry in the spacing at both ends shows that the gripper distance needs to be added to the state variables of the predictor, rather than assuming it is a constant property of the robot structure.

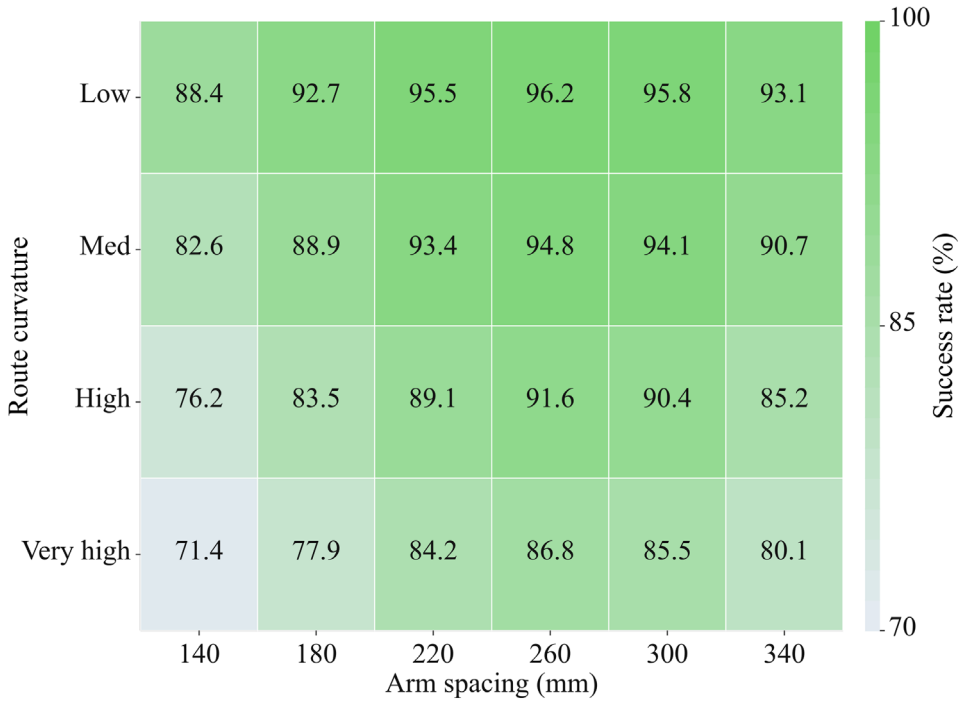


Figure 6. Routing success rate over dual-arm spacing and route curvature

Asynchronous perception updates can destabilize a dual-arm controller even when mean inference time is low [28]. Across 480 closed-loop replay trials, the measured latency profile did not cause a skipped control update. A supervisory rule limited robot end-effector speed to 0.72 m/s in high-bend-risk regions; severe-contact events decreased by 34.6%, while overall execution time increased by 5.8%. A second rule triggered replanning when predicted feasibility remained below 0.45 for five consecutive frames and prevented 71 of 85 manually identified failures. The controller therefore slowed the arms for high bend risk, requested a new observation for low contact confidence, and revised the waypoint when endpoint error persisted. In the double-corner case, the model linked nonadjacent cable segments to detect an approaching crossing, whereas during clamp insertion it emphasized the fixture throat and the opposite gripper. When depth layers were indistinguishable, however, none of the prediction heads could recover their ordering reliably.

6. Conclusion

SparseFormer-V2 is developed and tested in an experiment for cross-scale state prediction of dual-arm cable routing, showing that mechanically organized sparse attention can improve the prediction accuracy, robustness and computational efficiency under complex routing constraints.

1. SparseFormer-V2 achieved a routing-state accuracy of 92.8%, an endpoint prediction error of 5.1 mm, and a success-forecasting F1 score of 0.903. Compared with the dense transformer, the model reduced endpoint error by 40.7%, increased success F1 from 0.841 to 0.903, and shortened inference latency from 37.9 ms to 18.7 ms.

2. The model maintained clearer performance advantages as routing difficulty increased. At 45% visual occlusion, it retained 86.7% accuracy with a 7.8 mm endpoint error, whereas the dense transformer achieved 77.9% accuracy with a 12.6 mm error. Cross-scale attention was particularly beneficial in constrained trays, high-curvature routes, and long-slack conditions, while an average attention density of 8.7% reduced multiply-accumulate operations by approximately 46%.

3. The principal innovation is the integration of cable-centered tokens, route-contact anchors, dual-arm coordination states, and temporal evidence in one bounded sparse predictor. Its demonstrated applicability is limited to unbranched cables, visible active workspaces, fixed end grasps, known route geometry, and motion regimes in which inertial effects are secondary. Significant self-contact, deep fixture cavities, ambiguous depth ordering, and cable branching can remove the local evidence required by the attention anchors and therefore remain unresolved operating conditions.

Based on the above results, SparseFormer-V2 can serve as an efficient predictive module for closed-loop cable-routing systems, and in the future, tactile-visual fusion, branched harness representation, and long-horizon planning will be integrated with adaptive dual-arm impedance control.

7. Declarations

7.1 Author Contributions

Tomislav Matijaš contributes to conceptualization, methodology, software, validation, analysis, investigation, data collection, draft preparation, manuscript editing, visualization, supervision, project administration, and funding acquisition. Damir Zec contributes to validation, analysis, investigation, data collection, draft preparation. Robert Čurić contributes to conceptualization, validation, analysis. All authors have read and agreed with the manuscript before its submission and publication.

7.2 Funding and Acknowledgments

This research received no specific financial support from any funding agency.

7.3 Institutional Review Board Statement

Not applicable.

7.4 Informed Consent Statement

Not applicable.

7.5 Data Availability

Not applicable.

7.6 Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

7.7 Declaration of generative AI and AI-assisted technologies

During the preparation of this manuscript, the authors utilized Quillbot for linguistic polishing, logical organization and draft revision. The generative AI tool was not involved in research design, data collection, data analysis, or the equation of core conclusions. All outputs generated by AI were reviewed, revised and verified manually by the authors. The authors take full responsibility for all content of this manuscript.

8. References

[1] Zhou, P., Zheng, P., Qi, J., Li, C., Lee, H.-Y., Duan, A., Lu, L., Li, Z., Hu, L., & Navarro-Alarcon, D. (2024). Reactive human-robot collaborative manipulation of deformable linear objects using a new topological latent control model. Robotics and Computer-Integrated Manufacturing, 88, 102727. https://doi.org/10.1016/j.rcim.2024.102727

[2] Almaghout, K., Cherubini, A., & Klimchik, A. (2024). Robotic co-manipulation of deformable linear objects for large deformation tasks. Robotics and Autonomous Systems, 175, 104652. https://doi.org/10.1016/j.robot.2024.104652

[3] Szymko, A., Kicki, P., & Walas, K. (2024). Calibrationless bimanual deformable linear object manipulation with recursive least squares filter. IEEE Access, 12, 126707-126716. https://doi.org/10.1109/ACCESS.2024.3438624

[4] Wu, S., Zhang, J., & Wu, D. (2024). An optimization-based motion planner for dual-arm manipulation of the soft deformable linear objects with nonnegligible gravity. Advanced Engineering Informatics, 62, 102874. https://doi.org/10.1016/j.aei.2024.102874

[5] Cao, B., Zang, X., Zhang, X., Chen, Z., Li, S., & Zhao, J. (2024). Shape control of elastic deformable linear objects for robotic cable assembly. Advanced Intelligent Systems, 6(7), 2300835. https://doi.org/10.1002/aisy.202300835

[6] Almaghout, K., & Klimchik, A. (2024). Manipulation planning for cable shape control. Robotics, 13(1), 18. https://doi.org/10.3390/robotics13010018

[7] Caporali, A., Kicki, P., Galassi, K., Zanella, R., Walas, K., & Palli, G. (2024). Deformable linear objects manipulation with online model parameters estimation. IEEE Robotics and Automation Letters, 9(3), 2598-2605. https://doi.org/10.1109/LRA.2024.3357310

[8] Tang, Y., Chu, X., Huang, J., & Au, K. W. S. (2024). Learning-based MPC with safety filter for constrained deformable linear object manipulation. IEEE Robotics and Automation Letters, 9(3), 2877-2884. https://doi.org/10.1109/LRA.2024.3362643

[9] Monguzzi, A., Dotti, T., Fattorelli, L., Zanchettin, A. M., & Rocco, P. (2025). Optimal model-based path planning for the robotic manipulation of deformable linear objects. Robotics and Computer-Integrated Manufacturing, 92, 102891. https://doi.org/10.1016/j.rcim.2024.102891

[10] Wang, M., Wang, J., Zhang, J., Liao, X., & Li, G. (2025). Physics-informed graph learning for shape prediction in robot manipulate of deformable linear objects. Chinese Journal of Mechanical Engineering, 38, 161. https://doi.org/10.1186/s10033-025-01299-3

[11] Chen, Z., Huang, J.-A., Röning, J., Angrisani, L., & Li, S. (2025). Online iterative learning enhanced sim-to-real transfer for efficient manipulation of deformable objects. Machine Intelligence Research, 22(4), 696-712. https://doi.org/10.1007/s11633-025-1566-0

[12] Nicola, G., Villagrossi, E., & Pedrocchi, N. (2024). Co-manipulation of soft-materials estimating deformation from depth images. Robotics and Computer-Integrated Manufacturing, 85, 102630. https://doi.org/10.1016/j.rcim.2023.102630

[13] Wang, J., Xiong, X., Tolu, S., & Gorb, S. N. (2024). A learning-based control framework for fast and accurate manipulation of a flexible object. Journal of Bionic Engineering, 21(4), 1761-1774. https://doi.org/10.1007/s42235-024-00534-2

[14] Shneor, R., Naveh, G., Ben-David, S., Shvarzman, B., Mann, Z., Greenberg, A., Efrat, Y., Einav, O., & Berman, S. (2026). A planning and execution framework for robotic assembly with deformable objects using a centralized database based on the Rαβγ categorization. Journal of Intelligent Manufacturing, 37(2), 955-975. https://doi.org/10.1007/s10845-025-02578-5

[15] Wittmann, J., Ochsenfarth, F., Sonneville, V., & Rixen, D. (2024). Centralized vs. decoupled dual-arm planning taking into account path quality. Journal of Intelligent & Robotic Systems, 110, 141. https://doi.org/10.1007/s10846-024-02175-3

[16] D'Ago, G., Selvaggio, M., Suárez, A., Gañán, F. J., Buonocore, L. R., Di Castro, M., Lippiello, V., Ollero, A., & Ruggiero, F. (2024). Modelling and identification methods for simulation of cable-suspended dual-arm robotic systems. Robotics and Autonomous Systems, 175, 104643. https://doi.org/10.1016/j.robot.2024.104643

[17] Di Leva, R., Gattringer, H., Müller, A., & Carricato, M. (2024). Force and time-optimal trajectory planning for dual-arm unilateral cooperative grasping. Mechanism and Machine Theory, 201, 105729. https://doi.org/10.1016/j.mechmachtheory.2024.105729

[18] Zhou, Y., Li, Z., Feng, A., Zhao, W., & Zhu, M. (2025). Human-robot collaborative assembly of dual-arm manipulators using a four-level control framework. Nonlinear Dynamics, 113, 18243-18264. https://doi.org/10.1007/s11071-025-11106-2

[19] Liu, Y., Liu, S., Chen, B., Yang, Z.-X., & Xu, S. (2025). Fusion-perception-to-action transformer: Enhancing robotic manipulation with 3-D visual fusion attention and proprioception. IEEE Transactions on Robotics, 41, 1553-1567. https://doi.org/10.1109/TRO.2025.3539193

[20] Yao, X., Blei, T., Meng, Y., Zhang, Y., Zhou, H., Bing, Z., Huang, K., Sun, F., & Knoll, A. (2025). Long-horizon language-conditioned imitation learning for robotic manipulation. IEEE/ASME Transactions on Mechatronics, 30(6), 5628-5639. https://doi.org/10.1109/TMECH.2025.3547047

[21] Zhang, X., Liu, Y., Chang, H., Schramm, L., & Boularias, A. (2025). Autoregressive action sequence learning for robotic manipulation. IEEE Robotics and Automation Letters, 10(5), 4898-4905. https://doi.org/10.1109/LRA.2025.3550849

[22] Zhang, Y., Wu, R., Dascalu, S. M., & Harris, F. C., Jr. (2024). Multi-scale transformer pyramid networks for multivariate time series forecasting. IEEE Access, 12, 14731-14741. https://doi.org/10.1109/ACCESS.2024.3357693

[23] Li, A., Li, Y., Xu, Y., Li, X., & Zhang, C. (2024). Multi-scale convolution enhanced transformer for multivariate long-term time series forecasting. Neural Networks, 180, 106745. https://doi.org/10.1016/j.neunet.2024.106745

[24] Chen, Y., Liu, S., Yang, J., Jing, H., Zhao, W., & Yang, G. (2024). A joint time-frequency domain transformer for multivariate time series forecasting. Neural Networks, 176, 106334. https://doi.org/10.1016/j.neunet.2024.106334

[25] Tian, G., Zhang, C., Shi, Y., & Li, X. (2024). MultiWaveNet: A long time series forecasting framework based on multi-scale analysis and multi-channel feature fusion. Expert Systems with Applications, 251, 124088. https://doi.org/10.1016/j.eswa.2024.124088

[26] Zhang, Y., Wu, R., Dascalu, S. M., & Harris, F. C., Jr. (2024). Sparse transformer with local and seasonal adaptation for multivariate time series forecasting. Scientific Reports, 14, 15909. https://doi.org/10.1038/s41598-024-66886-1

[27] Gao, L., Gu, X., Chen, F., & Wang, J. (2024). Sparse transformer network with spatial-temporal graph for pedestrian trajectory prediction. IEEE Access, 12, 144725-144737. https://doi.org/10.1109/ACCESS.2024.3442442

[28] Caporali, A., Galassi, K., & Palli, G. (2024). DLO Perceiver: Grounding large language model for deformable linear objects perception. IEEE Robotics and Automation Letters, 9(12), 11385-11392. https://doi.org/10.1109/LRA.2024.3491428