Dear Reviewer DGZV,

Thank you for your detailed comments and suggestions. We tried our best to address all the concerns and questions, and update the main paper and appendix(marked red) in the new version. Please let us know if you have any further concerns or questions to discuss.

Best,

Paper 1571 Authors

W1. About the motivation of the interpolation and more experiments for upper bound.

A1.

Motivation&Challenge:

- **Task**: Our method addresses a fundamental challenge in instructional video procedure planning - predicting intermediate actions when only given start and end visual information. Since we lack direct supervision for intermediate states during training and inference, developing an effective approach to *reconstruct* these missing visual features becomes crucial for accurate prediction.
- **Supervision level**: While existing approaches are constrained by text-level supervision that offers only limited insights into visual state transitions, our method advances beyond this by incorporating *visual-level* supervision from start and end observations. However, we face a critical challenge - the absence of real intermediate visual features during the denoising process. This limitation motivates our novel interpolation approach that reconstructs these missing visual states, enabling us to model the temporal progression in significantly more detail and comprehensiveness compared to methods relying solely on text supervision.
- **Interpolation**: We propose interpolation as an elegant and effective solution to generate intermediate visual features, addressing the absence of real intermediate visual features especially during inference. Visual features inherently possess continuity and logical coherence, allowing interpolation to naturally preserve the holistic semantic information while introducing rich and diverse details. Furthermore, the mathematical simplicity and computational efficiency of interpolation make it an ideal choice that seamlessly integrates into our framework.

However, this task remains challenging. As shown in Table 6 (ablation experiments for our latent space temporal logical interpolation module), using only an interpolator does not achieve the best performance. To address this, we have implemented several enhancements:

- We added an *observation encoder* before the interpolator to extract latent features and filter out noise.
- We incorporated *transformer encoder blocks* to enhance logical relationships between features.

These additions have significantly improved our model's performance, leading to the best results among compared methods.

Upper Bound: As suggested, we have conducted additional experiments using real visual features as supervision to test the upper bound of our method. The outcomes vary based on dataset characteristics (size, especially the kinds of tasks and actions, and average action sequence lengths). To explain this phenomenon, we divide our interpolated features into two parts: *simple memory* and *hard temporal logical relationships*.

For COIN, with its largest size but shortest sequences, interpolated features excel in simple memory-focused tasks. NIV, the smallest but with longest sequences, shows comparable performance between real and interpolated features. CrossTask, being large with long sequences, reveals a significant performance gap favoring real features.

These findings highlight a trade-off: interpolated features perform well in simpler datasets but struggle with complex temporal relationships in larger, more diverse datasets. This underscores the need for improved interpolation methods to effectively handle complex, temporally diverse datasets in future work.

Table 1: Upper bound of visual features supervision.

Dataset	Method	SR↑ (T=3)	mAcc↑ (T=3)	mIoU↑ (T=3)	SR↑ (T=4)	mAcc↑ (T=4)	mIoU↑ (T=4)	SR↑ (T=5)	SR↑ (T=6)
CrossTask	Interpolated	40.45	67.19	69.17	24.76	60.69	67.67	15.26	10.30
	Real	**49.05**	**73.62**	**73.23**	**36.55**	**70.42**	**72.09**	**24.88**	**24.02**
COIN	Interpolated	**30.90**	**52.17**	**59.58**	**23.10**	**49.71**	**60.78**	-	-
	Real	27.07	49.07	57.53	20.01	47.35	58.24	-	-
NIV	Interpolated	29.63	48.02	**56.49**	**25.76**	46.62	58.50	-	-
	Real	**32.59**	**50.25**	56.40	24.02	**48.36**	**58.92**	-	-

Revision. In the revision, we have added the discussion of upper bound of visual features supervision in Appendix D under the section "Upper Bound of Visual Features Supervision" at line 1000.

W2. A MLP is enough for this simple classification task.

A2. For long instructional videos, MLP classifiers are too simple to model complex spatial-temporal relationships between actions to obtain accurate task labels. To address this limitation, we propose transformer-based classifiers, which can better capture contextual relationships between classes through their self-attention mechanisms and long-term temporal relationship modeling capabilities. This is particularly important since classification accuracy is a crucial factor in the overall performance of our method.

To validate this, we conducted comprehensive ablation studies on classifier types in Appendix C (see "Transformer Classifier Type" section at line 863). Our experimental results (two tables below) show that using a better-performing transformer classifier leads to significant improvements in the overall results themselves under the same conditions.

Table 2: Classification results on CrossTask.

Models	T=3	T=4	T=5	T=6
Res-MLP(PDPP)	92.43	92.98	93.39	93.20
Transformer(Ours)	**93.67**	**94.03**	**94.02**	**94.26**

Table 3: Performance comparison on CrossTask.

Model	SR↑	mAcc↑	mIoU↑
PDPP (Res-MLP)	37.2	55.35	66.57
PDPP (Transformer)	39.08	66.32	68.47
MTID (Transformer)	**40.45**	**67.19**	**69.17**

Moreover, compared to PDPP's MLP and our transformer-based classifier, here is the resource comparison on CrossTask (with GeForce RTX 4090):

Table 4: Resource comparison on CrossTask.

Model Type	Memory Usage	Training Time
PDPP (MLP)	1237M	21min

Model Type	Memory Usage	Training Time
Ours (Transformer)	1637M	25min

Given the significant performance improvements shown in Table 2&3, we believe this slight increase in computational resources is a worthwhile **trade-off**.

W3. The masked projection introduces too much prior.

A3. The masked projection serves as a training-time constraint that helps guide and restrict the action space during initialization. Rather than enforcing a rigid mask prior at inference time, it acts as a soft guidance mechanism during model training.

Our experimental results demonstrate the effectiveness of this approach. As shown in the table below, even without utilizing masked projection (MP), our method still outperforms other compared approaches on the CrossTask dataset, highlighting the robustness of our overall methodology.

Table 5: Performance comparison on CrossTask.

Method	SR↑	mAcc↑
KEPP	38.12	64.74
SCHEMA	38.93	63.80
MTID w/o MP	**39.17**	**66.49**

W4. Details about the weighted gradient loss and analyze the loss ablation results.

A4.

Analysis for ablation of loss: We explain these results (where the improvement from ID 3 to 6 is only 0.16) by noting that our mask not only filters irrelevant actions but also provides weight constraints on intermediate actions. However, since the mask applies equal weights to all intermediate timesteps, its effectiveness is not as strong as GW which applies gradient weights. This explains why using both together (mask and GW) shows improvement but the gain is relatively small due to their partially overlapping effects.

This explanation is supported by evidence from the improvements seen from ID 1 to 3 and ID 1 to 4. Both methods show approximately 2-point improvements: The mask increases performance from 11.89 (ID1) to 13.26 (ID4), while gradient weights improve performance from 11.89 (ID1) to 13.90 (ID2). These results demonstrate that both our mask and gradient weights independently improve model performance effectively. When used together, although the improvement is modest, it is still noticeable.

Table 6: Ablation studies on our loss function. Note: W: Weights on Both Sides, GW: Gradient Weights, M: Mask.

ID	MSE	GW	M	SR↑
1	✓			11.89
3	✓	✓		15.10
4	✓		✓	13.26
6	✓	✓	✓	**15.26**

Explanation for weighted gradient loss: In our task-adaptive masked proximity loss,

$$ \mathcal{L}_{\mathrm{diff}} = \sum_{t=1}^{T} \sum_{d=1}^{A} w_t \cdot m_{t,d} \cdot (a_{t,d} - \bar{a}_{t,d})^2,$$

$$w_t = w_0 + (1 - w_0) \cdot \frac{\min(t, T - t + 1) - 1}{\lceil T/2 \rceil - 1},$$

the weight w_t linearly decreases from both ends toward the middle in a slope-like pattern, so that is why we refer to them as gradient weights in our experiments. The gradient weights are combined with MSE to form the weighted gradient loss, emphasizing the gradual change of weights across different timesteps.

> **Note:** Weighted Loss and Weighted Gradient Loss are two different weighting strategies, and they can not be used together. Therefore, in ID 6 we only use MSE+GW+M.
>
> ---

W5. The novelty is limited compared with PDPP.

A5. Our method tackles a fundamental challenge in instructional video procedure planning: accurately predicting the sequence of intermediate actions given only the start and end visual states. This complex task demands sophisticated capabilities in both temporal prediction and logical reasoning. While the existing PDPP approach employs a basic U-Net architecture with diffusion-based prediction, it falls short in capturing the intricate relationships between actions. In contrast, our method introduces several innovative components that significantly enhance the modeling of temporal and logical dependencies between actions:

1. **Supervision-level Innovation**:

    - Challenge: Existing approaches, particularly PDPP, primarily utilize text-level supervision, which inherently limits their ability to capture the rich visual dynamics and state transitions between actions.
    - Our Solution: We introduce a novel visual-level supervision paradigm that leverages both start and end state observations. Through our innovative interpolation mechanism, we effectively reconstruct intermediate visual features that would otherwise be missing. This approach enables our model to capture temporal progression with significantly greater fidelity compared to traditional text-based supervision methods.

2. **Architecture-level Innovations**:

    a) **Latent Space Temporal Logical Interpolation Module**:

    - Challenge: PDPP's simple U-Net architecture lacks the sophistication needed to effectively model and capture the complex temporal relationships and logical dependencies between sequential actions.
    - Our Improvement: We propose a novel interpolation module that intelligently reconstructs missing intermediate visual features, significantly enhancing the model's ability to reason about temporal progression and seamlessly integrate features across different timesteps.

    b) **Task-adaptive Masked Projection**:

    - Challenge: PDPP's design leads to predictions more likely to fall outside the current task's action space.
    - Our Improvement: We implement an adaptive mechanism to constrain the action space during initialization by masking out irrelevant actions for current task, reducing out-of-scope predictions.

3. **Loss-level Innovation**:

    - Challenge: PDPP's loss function causes predictions more likely to fall outside the task's action space and overemphasizes endpoints.

    - Our Improvements:

        a) **Gradient-guided weighting**: Balances supervision across all timesteps while maintaining emphasis on endpoints.

        b) **Task-specific mask mechanism**: Further limits the action space to align with task-specific constraints.

4. **Efficiency Improvements**:

    - Challenge: PDPP's DDPM diffusion process is time-consuming.

- Our Improvement: We replace it with DDIM, accelerating both training and inference while maintaining prediction quality.

5. **Task Classification**:

   - Challenge: PDPP's simple MLP classifier is insufficient for complex task classification.
   - Our Improvement: We enhance it with a transformer module, improving complex task classification through better modeling of long-term temporal relationships.

These innovations work synergistically to significantly advance the state-of-the-art across benchmarks (CrossTask: +3 points, COIN: +9 points compared to PDPP), demonstrating substantial improvements over PDPP's framework.