A Stitch in Time: Learning Procedural Workflow via Self-Supervised Plackett–Luce Ranking

Supplementary Material

A. Method Details

We provide the comprehensive mathematical formulation of our Plackett-Luce ranking framework and describe its implementation across the proposed video and image branches.

A.1. Plackett-Luce Ranking Formulation

Our framework leverages the **Plackett-Luce (PL) model** [12, 15] to structure both the global and local objectives as listwise ranking problems. The PL model is parameterized by a vector of positive real-valued scores $s = (s_1, \dots, s_K)$, where K is the number of items being ranked. The score s_i represents the overall utility or preference strength of item i.

The probability $P(r|s)$ of observing a specific full ranking (permutation) $r = (r(1), r(2), \dots, r(K))$, where r is a reordering of $\{1, \dots, K\}$, is defined sequentially based on Luce's Choice Axiom [12]. Specifically, the probability is the product of the conditional probabilities of choosing item $r(i)$ from the set of items R_i not yet ranked at step i:

$$P(r|s) = \prod_{i=1}^{K} \frac{\exp(s_{r(i)})}{\sum_{j=i}^{K} \exp(s_{r(j)})} \tag{1}$$

The model is trained by minimizing the negative log-likelihood of the single ground-truth permutation $r^* = (r^*(1), \dots, r^*(K))$. The loss function $\mathcal{L}_{\text{PL}}$ is defined as:

$$\mathcal{L}_{\text{PL}}(s, r^*) = -\log P(r^*|s) \tag{2}$$

By applying the negative logarithm to the product in the PL definition, the loss $\mathcal{L}_{\text{PL}}$ unfolds into a sum of K distinct preference decisions. This fully expanded form is minimized during training:

$$\mathcal{L}_{\text{PL}}(s, r^*) = -\sum_{i=1}^{K} \left[\log\left(\exp(s_{r^*(i)})\right) - \log\left(\sum_{j=i}^{K} \exp(s_{r^*(j)})\right) \right] \tag{3}$$

This simplifies to the final loss form:

$$\mathcal{L}_{\text{PL}}(s, r^*) = \sum_{i=1}^{K} \left[\log\left(\sum_{j=i}^{K} \exp(s_{r^*(j)})\right) - s_{r^*(i)} \right] \tag{4}$$

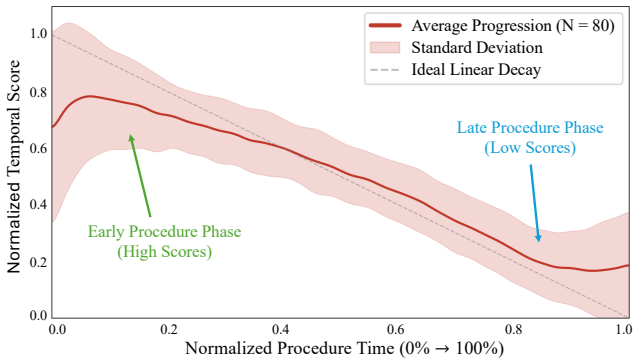


Figure 1. **Global Procedural Progression.** We visualize the temporal progression score averaged across all 80 Cholec80 [18] videos, where higher scores indicate earlier phases. To generate this plot, we extract [CLS] embeddings using the frozen backbone of our trained PL-Stitch model and predict frame-wise scores using its temporal head. These scores are normalized per video to a unit scale ($0 \rightarrow 1$) before computing the mean (solid line) and standard deviation (shaded region). Despite being trained on the different dataset, LEMON [3], the model successfully generalizes to the unseen Cholec80 videos. The predicted score consistently decreases during the active surgical workflow. The observed deviations at the boundaries ($t < 0.1$ and $t > 0.9$) correspond to the camera entering and exiting the body, effectively marking the non-operative transitions surrounding the procedure.

where i indexes the rank position being determined, $r^*(i)$ denotes the item at rank i in the ground-truth sequence r^*, and the inner summation over j computes the total score of all items remaining in the unranked positions from i to K.

This compels the encoder to assign a significantly higher score $s_{r^*(i)}$ to the correct frame (or patch) chosen at step i compared to the log-sum-exponent of all remaining items. This probabilistic approach scales the error penalty proportionally to the ranking mistake severity, proving more robust than permutation classification [6, 13, 21, 22] or pairwise losses [8, 10, 16].

A.2. Video Branch: Listwise Temporal Ranking

The Video Branch implements the global workflow progression objective $\mathcal{L}_{\text{vid}}$ as an instantiation of the $\mathcal{L}_{\text{PL}}$ loss.

- **Items (K):** A clip $C_v = (v_1, \dots, v_K)$ of $K = 8$ sampled frames is used. The PL parameters $s_{\text{clip}} = (s_1, \dots, s_K)$ are predicted from the [CLS] tokens of the frames via the temporal head h_{vid}.
- **Ground-Truth (r^*):** The target permutation is the true

chronological order: $r^*_{\text{clip}} = (1, 2, \ldots, K)$.

- **Loss Function:** The objective minimizes the difference between the predicted scores and the true temporal order:

$$\mathcal{L}_{\text{vid}} = \mathcal{L}_{\text{PL}}(s_{\text{clip}}, r^*_{\text{clip}}).$$

A.3. Image Branch: Spatio-temporal Jigsaw

The Image Branch implements the $\mathcal{L}_{\text{jigsaw}}$ objective, another instantiation of the $\mathcal{L}_{\text{PL}}$ loss, designed to capture fine-grained object correspondence using local temporal context, inspired by [7, 11].

- **Items (N):** The items are the N patches of the central, masked frame v_t.
- **Temporal Context Injection:** Patches of the masked current frame serve as **Queries (Q)** ($\tilde{E}^{\text{patch}}$). The concatenated embeddings from adjacent **past** ($v_{t-\tau_1}$) and **future** ($v_{t+\tau_2}$) frames serve as **Keys (K) and Values (V)**, forcing the model to rely on temporal consistency for spatial reconstruction (see Sec. B.1 for τ_1 and τ_2 sampling details).
- **Ground-Truth (r^*):** The target permutation is the original linearized patch order: $r^*_{\text{jigsaw}} = (1, 2, \ldots, N)$.
- **Loss Function:** The jigsaw head h_{jigsaw} predicts s_{jigsaw}, which is minimized using:

$$\mathcal{L}_{\text{jigsaw}} = \mathcal{L}_{\text{PL}}(s_{\text{jigsaw}}, r^*_{\text{jigsaw}}).$$

A.4. Learned Global Workflow Progression

To investigate whether PL-Stitch captures the global procedural structure, we conducted a qualitative analysis on the Cholec80 [18] dataset. We employed the model pretrained on the large-scale LEMON dataset [3], which contains no samples from Cholec80, ensuring a strict zero-shot evaluation. Specifically, we processed all 80 downstream videos by extracting [CLS] embeddings via the **frozen PL-Stitch backbone** and subsequently generating frame-wise real-valued scores using the **pretrained temporal head**.

Fig. 1 illustrates the temporal progression score averaged across all videos, with the procedural duration normalized to a unit scale ($0 \rightarrow 1$). Despite being pretrained with an 8-frame ranking objective, the model exhibits a remarkable capacity to generalize globally. The average score demonstrates a consistently decreasing trend over the course of the procedure. This behavior is a direct consequence of the Plackett-Luce optimization: to maximize the likelihood of the correct chronological order, the model is trained to assign larger scores to the earlier frames in any sampled sequence. This confirms that PL-Stitch has effectively learned to map the visual evolution of the procedure to a continuous scalar representation of procedural progress.

Notably, we observe a slight deviation from strict monotonicity at the extreme temporal boundaries (approx. $t < 0.1$ and $t > 0.9$). This behavior aligns with the visual semantics of the Cholec80 dataset. The initial "Preparation" and final "Retraction" phases share similar visual characteristics, as both involve the camera entering or exiting the body. These frames often depict blurry views of the abdominal wall or out-of-body scenes where no surgical tools are present. Consequently, the model assigns the peak "earliness" score not to these ambiguous pre-operative frames, but to the onset of the first active surgical phase ($t \approx 0.1$). This suggests that PL-Stitch goes beyond simple frame counting. It identifies the effective start of the operative workflow by recognizing the visual cues of active surgery while distinguishing them from non-informative idle states at the video boundaries.

B. Implementation Details

We provide the implementation specifics and hyperparameters for our pretraining framework.

B.1. Pretraining Details

We detail the implementation settings for our pretraining objective, which is the weighted sum $\mathcal{L}_{\text{total}} = \lambda_1\mathcal{L}_{\text{vid}} + \lambda_2\mathcal{L}_{\text{MIM}} + \lambda_3\mathcal{L}_{\text{jigsaw}}$.

- **Backbone and Optimizer:** We use a ViT-B/16 [4] backbone. The optimizer is AdamW, with a base learning rate of 4×10^{-4}, a weight decay of 0.05, and a total batch size of 240.
- **Pretraining Length:** Surgical models are pretrained on the LEMON dataset [3] for 30 epochs. Cooking models are pretrained directly on their respective official training sets for 100 epochs. These durations were specifically selected to guarantee that the models are trained sufficiently well and that the training loss has effectively converged for each domain.
- **Image-branch dataset construction:** We sample the pretraining videos at a rate of 1 fps to construct the dataset for the image branch.
- **Video-branch dataset construction:** We extract 8-frame clips to construct the dataset for the video branch. We sample multiple clips from each video, scaling the count with the video duration, to ensure the final size of the clip dataset is identical to that of the image branch.
- **Maximum iterations L**: This is calculated by dividing the total image-branch dataset size by the batch size, representing the number of steps required to complete one full epoch.
- **$\mathcal{L}_{\text{vid}}$ Parameters:** The clip length is set to $K = 8$ frames.
- **$\mathcal{L}_{\text{MIM}}$ Parameters:** A block-wise masking ratio of 30% is applied to the current frame v_t, following the iBOT protocol [23].
- **$\mathcal{L}_{\text{jigsaw}}$ parameters**: Following [11], which samples context frames from temporal offsets in the range $\pm[0.15T, 0.25T]$ on Kinetics-400 [2] videos (average duration $T \approx 10$ s), we randomly sample the temporal offsets τ_1 and τ_2 from the range $[1.5, 2.5]$ s relative to the

Table 1. **Ablation on loss weights** (λ).

No.	λ_1 ($\mathcal{L}_{\text{vid}}$)	λ_2 ($\mathcal{L}_{\text{MIM}}$)	λ_3 ($\mathcal{L}_{\text{jigsaw}}$)	Linear	k-NN
1	0.0	1.0	0.0	73.4	69.4
2	1.0	1.0	0.0	77.1	78.9
3	1.0	1.0	1.0	76.9	78.4
4	1.0	1.0	0.4	**77.8**	**80.2**
5	0.5	1.0	0.4	77.0	78.5
6	2.0	1.0	0.4	77.4	79.5

current frame.

B.2. Ablation on Loss Weights

We investigate the contribution of each objective and the sensitivity to loss weights on the Cholec80 phase recognition task. Results are reported in Table 1, where our optimal configuration is highlighted in gray. First, the addition of the temporal ranking loss $\mathcal{L}_{\text{vid}}$ (row 2) yields a substantial gain over the MIM-only baseline (row 1), boosting k-NN accuracy by **+9.5 pp** (from 69.4% to 78.9%). This confirms that explicit temporal ordering is the primary driver of procedural awareness.

Regarding the jigsaw weight λ_3, we observe that while removing it entirely (row 2) lowers accuracy, increasing it to $\lambda_3 = 1.0$ (row 3) also proves suboptimal (78.4% in k-NN). This suggests that while local spatio-temporal context is beneficial, an excessive jigsaw weight may distract the model from learning the global workflow progression. Finally, regarding the main temporal weight λ_1, our default value of 1.0 (row 4) outperforms both halving (row 5) and doubling (row 6) the weight, achieving the peak k-NN performance of 80.2%.

C. Downstream Task Details

We evaluate feature quality on five procedural video benchmarks for temporal phase recognition and action segmentation.

C.1. Evaluation Datasets

We evaluate our method on five challenging procedural benchmarks, covering both the surgical and cooking domains.

- **Cholec80 [18]**: This dataset consists of 80 videos of laparoscopic cholecystectomy procedures. The objective is surgical phase recognition, a frame-wise classification task where the model must assign one of 7 distinct surgical phases (e.g., Preparation, Calot-Triangle Dissection, Clipping and Cutting) to every frame in the video.
- **AutoLaparo [20]**: A dataset containing 21 videos of laparoscopic hysterectomy procedures. The task is surgical phase recognition, challenging the model to identify the current phase from 7 defined surgical phases across long, untrimmed videos.
- **M2CAI16 [17]**: This dataset features 41 videos of laparoscopic cholecystectomy. The task is surgical phase recognition, where the model must recognize 8 surgical phases.
- **Breakfast [9]**: A dataset comprising 1712 videos capturing 10 different breakfast preparation activities (e.g., making coffee, pancakes) from a third-person perspective. The task is temporal action segmentation, which involves densely classifying video frames into 48 fine-grained action steps (e.g., "pour milk", "crack egg").
- **GTEA [5]**: An egocentric (first-person) dataset containing 28 videos of daily cooking activities. The task is temporal action segmentation across 11 action classes, presenting unique challenges due to severe camera motion and hand occlusions typical of wearable cameras.

C.2. Evaluation Metrics

Linear probing. We evaluate the quality of the learned representations using the following standard metrics under the linear probing protocol.

- **Accuracy (Acc).** For all tasks, we report standard frame-wise top-1 accuracy. It is computed as the ratio of correctly classified frames to the total number of frames T:

$$\text{Acc} = \frac{1}{T}\sum_{t=1}^{T}\mathbb{I}(\hat{y}_t = y_t),$$

  where $\hat{y}_t$ is the predicted class for frame t, y_t is the ground-truth class, and $\mathbb{I}(\cdot)$ is the indicator function.
- **F1-score (F1).** For surgical phase recognition, we report the frame-wise macro F1-score, defined as the average of the per-class F1-scores. For each class c, the F1-score is the harmonic mean of precision and recall:

$$\text{F1}_c = 2 \cdot \frac{\text{Precision}_c \cdot \text{Recall}_c}{\text{Precision}_c + \text{Recall}_c},$$

  with

$$\text{Precision}_c = \frac{TP_c}{TP_c + FP_c}, \quad \text{Recall}_c = \frac{TP_c}{TP_c + FN_c},$$

  where TP_c, FP_c, and FN_c denote the numbers of true positives, false positives, and false negatives for class c, respectively. The reported macro F1-score is then

$$\text{F1}_{\text{macro}} = \frac{1}{C}\sum_{c=1}^{C}\text{F1}_c,$$

  where C is the total number of classes.
- **Segmental F1@δ.** For cooking action segmentation, we evaluate the quality of predicted segments using the segmental F1-score at Intersection over Union (IoU) thresholds $\delta \in \{10\%, 25\%, 50\%\}$. A predicted segment is counted as a true positive ($TP@\delta$) if its IoU with a

Table 2. **Five-fold cross-validation results on surgical datasets.** We report the mean ± std of the linear probing accuracy.

Method	AutoLaparo	Cholec80	M2CAI16
VideoMAEv2 [19]	50.3 ± 1.9	56.5 ± 1.7	51.4 ± 3.4
DINO [1]	75.5 ± 1.6	73.5 ± 1.5	68.9 ± 2.3
iBOT [23]	75.6 ± 2.2	75.9 ± 1.2	71.5 ± 2.1
PL-Stitch (Ours)	$\mathbf{80.1 \pm 2.5}$	$\mathbf{82.6 \pm 1.8}$	$\mathbf{75.2 \pm 2.0}$

ground-truth segment of the same class exceeds δ (and each ground-truth segment is matched to at most one prediction). The segmental F1@δ is defined as

$$\text{F1@}\delta = 2 \cdot \frac{\text{Precision@}\delta \cdot \text{Recall@}\delta}{\text{Precision@}\delta + \text{Recall@}\delta},$$

where

$$\text{Precision@}\delta = \frac{TP@\delta}{N_{\text{pred}}}, \quad \text{Recall@}\delta = \frac{TP@\delta}{N_{\text{gt}}},$$

and N_{pred} and N_{gt} are the numbers of predicted and ground-truth segments, respectively.

- **Edit distance (Edit).** For cooking tasks, we also measure temporal ordering consistency using a normalized Levenshtein edit score. Let S_{pred} and S_{gt} denote the predicted and ground-truth sequences of action segments, respectively. The edit score is defined as

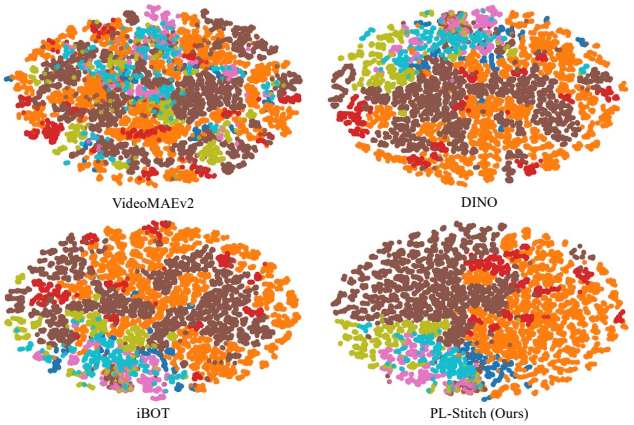
$$\text{Edit} = \left(1 - \frac{\text{lev}(S_{\text{pred}}, S_{\text{gt}})}{\max(|S_{\text{pred}}|, |S_{\text{gt}}|)}\right) \times 100,$$

where $\text{lev}(\cdot,\cdot)$ is the Levenshtein distance between two sequences and $|\cdot|$ denotes sequence length (number of segments).

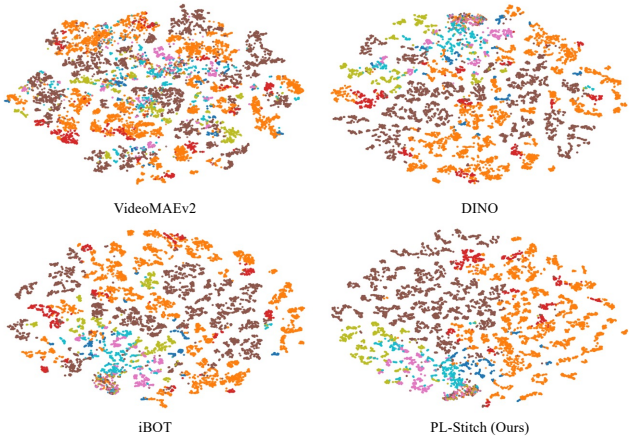
***k*-NN evaluation.** Following established self-supervised learning benchmarks [1, 14, 23], we assess the quality of the frozen feature space using non-parametric k-Nearest Neighbor (k-NN) classification. We first extract features from the frozen backbone for all frames in the training and testing sets. For each test frame, we retrieve its $k = 20$ nearest neighbors from the training set based on cosine similarity. The predicted class is determined via weighted majority voting, where neighbors are weighted by their similarity scores. This protocol provides a direct measure of the semantic separability of the learned representation without requiring parameter updates. We report the top-1 accuracy.

D. Additional Experimental Results

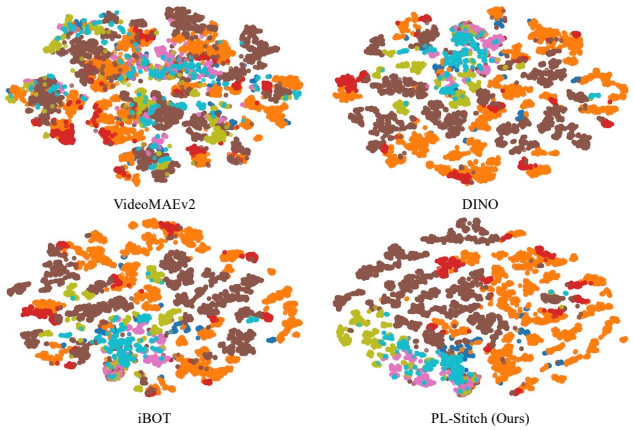
We present further quantitative and qualitative analyses to demonstrate the statistical robustness and semantic interpretability of the representations learned by PL-Stitch.



(a) Perplexity = 10 (Local Structure)



(b) Perplexity = 30 (Balanced / Default)



(c) Perplexity = 50 (Global Structure)

Figure 2. **Robustness of feature embeddings on the Cholec80 dataset under varying t-SNE parameters.** Comparison of feature visualizations at (a) Perplexity 10, (b) the default Perplexity 30, and (c) Perplexity 50. While baselines show mixed clusters across all settings, PL-Stitch consistently maintains clearer class separation.

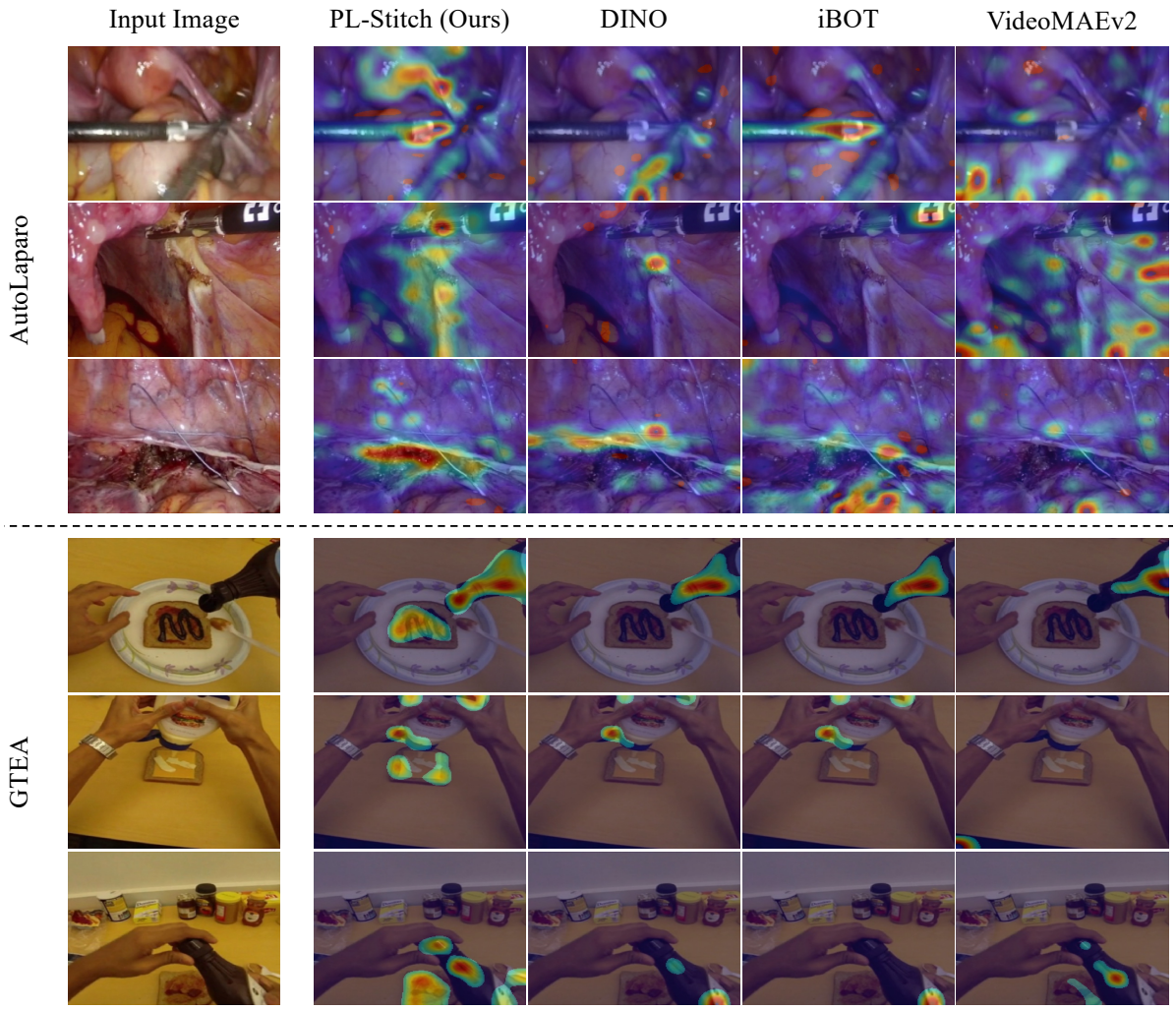


Figure 3. **Qualitative comparison of attention localization across diverse procedural scenes.** We visualize the attention maps queried by the [CLS] token for our method (PL-Stitch) and other models (DINO, iBOT, VideoMAEv2) on input images from AutoLaparo (top) and GTEA (bottom). PL-Stitch consistently demonstrates a superior ability to localize and focus its attention on key interaction areas, such as surgical instruments or manipulated objects. This outperforms other methods that exhibit more diffuse or misplaced attention.

D.1. Five-Fold Cross-Validation on Surgical Phase Recognition

In the surgical domain, robust evaluation is critical to ensure that models generalize effectively across varying patient anatomies and surgical workflows. To verify the statistical robustness of our method in this challenging setting, we performed 5-fold cross-validation across all three surgical datasets: Cholec80 [18], AutoLaparo [20], and M2CAI16 [17]. We report the linear probing top-1 accuracy (Mean $\pm$ Std) to demonstrate the stability of the learned features.

As shown in Table 2, PL-Stitch yields the highest mean accuracy across all datasets while maintaining a low standard deviation. This confirms that the procedurally-aware representations learned by our model are not only discriminative but also highly stable across different data splits and surgical domains.

D.2. Sensitivity Analysis of t-SNE Visualization

In the qualitative evaluation presented in the main manuscript, we employed a default t-SNE perplexity of 30, which offers a balanced representation of both local and global feature structures. To verify that the observed class separability is an intrinsic property of the learned embeddings rather than an artifact of visualization parameter tuning, we provide a robustness analysis on the Cholec80 dataset [18] in Fig. 2.

We visualize feature embeddings at perplexities 10, 30, and 50 (Figs. 2a, 2b, 2c). At perplexity 10, the focus on lo-

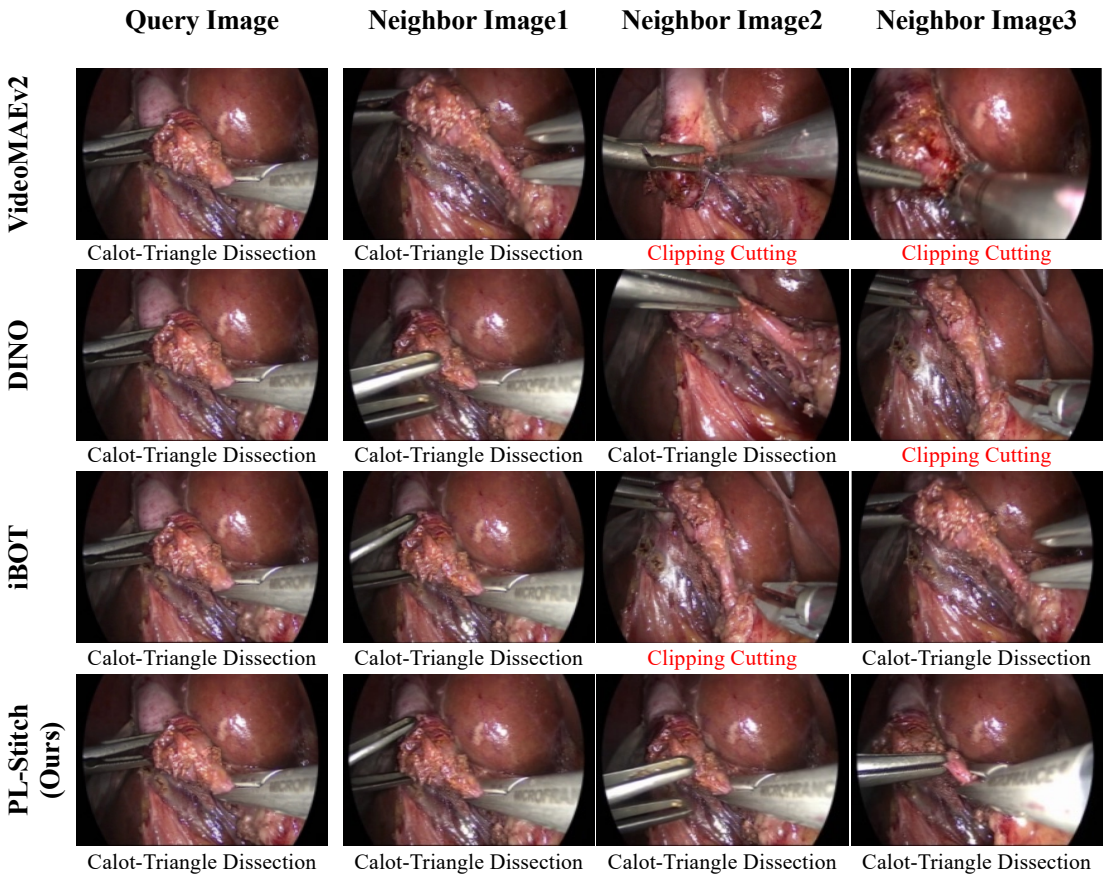


Figure 4. **Nearest Neighbor Retrieval.** Comparison of the top-3 retrieved frames for a query image from the Calot-Triangle Dissection phase. Incorrect phase predictions are highlighted in red text. Baselines such as VideoMAEv2, DINO, and iBOT are deceived by visual similarity and incorrectly retrieve frames from the Clipping Cutting phase. PL-Stitch retrieves only procedurally synchronous frames, unlike baselines which fail to distinguish between similar-looking but distinct procedural steps.

cal neighborhoods causes fragmentation, yet PL-Stitch retains identifiable groupings. Increasing the perplexity to 30 and 50 reveals distinct and well-separated clusters for our method as global structure becomes emphasized. Conversely, VideoMAEv2, DINO, and iBOT show persistent overlap between similar phases across all settings. This consistency confirms the robustness of our learned feature space.

D.3. Additional Attention Maps

We provide an extended qualitative analysis of the attention focus of the model by visualizing self-attention maps queried by the [CLS] token across diverse procedural contexts in Fig. 3. This comparison encompasses both surgical scenes from the AutoLaparo dataset [20] and cooking activities from the GTEA dataset [5]. PL-Stitch consistently concentrates high attention weights within task-relevant areas and demonstrates a strong semantic alignment with the workflow. For instance, in the AutoLaparo examples, our PL-Stitch model's attention remains anchored on the instrument-tissue interaction sites and demonstrates robust tracking of the surgical flow. Similarly, in the GTEA examples, attention accurately tracks the manipulated objects and active interaction zones, such as the condiment container and the spread on the bread. In contrast, baseline methods such as DINO, iBOT, and VideoMAEv2 exhibit significantly more diffuse attention patterns that often drift towards background elements or fail to distinctly highlight the active interaction site. This comparison underscores the stability and precision of PL-Stitch in localizing key visual cues compared to prior self-supervised approaches.

D.4. Semantic Feature Retrieval

Fig. 4 shows a nearest-neighbor retrieval comparison on the Cholec80 dataset. Given a query image, baseline models frequently retrieve images that appear visually similar but belong to the wrong procedural phase. In contrast, PL-Stitch correctly retrieves images only from the correct phase and demonstrates a robust understanding of the underlying procedural workflow.

References

[1] Mathilde Caron, Hugo Touvron, Ishan Misra, Herve Jegou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging Properties in Self-Supervised Vision Transformers. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9630–9640. IEEE, 2021. 4

[2] Joao Carreira and Andrew Zisserman. Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4724–4733. IEEE, 2017. 2

[3] Chengan Che, Chao Wang, Tom Vercauteren, Sophia Tsoka, and Luis C. Garcia-Peraza-Herrera. LEMON: A Large Endoscopic MONocular Dataset and Foundation Model for Perception in Surgical Settings. *arXiv preprint arXiv:2503.19740*, 2025. Accepted at CVPR 2026. 1, 2

[4] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. *arXiv preprint arXiv:2010.11929*, abs/2010.11929, 2020. 2

[5] Alireza Fathi, Xiaofeng Ren, and James M. Rehg. Learning to recognize objects in egocentric activities. In *CVPR 2011*, pages 3281–3288. IEEE, 2011. 3, 6

[6] Basura Fernando, Hakan Bilen, Efstratios Gavves, and Stephen Gould. Self-Supervised Video Representation Learning with Odd-One-Out Networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5729–5738. IEEE, 2017. 1

[7] Agrim Gupta, Jiajun Wu, Jia Deng, and Li Fei-Fei. Siamese masked autoencoders. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, Red Hook, NY, USA, 2023. Curran Associates Inc. 2

[8] Kai Hu, Jie Shao, Yuan Liu, Bhiksha Raj, Marios Savvides, and Zhiqiang Shen. Contrast and Order Representations for Video Self-supervised Learning. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 7919–7929. IEEE, 2021. 1

[9] Hilde Kuehne, Ali Arslan, and Thomas Serre. The Language of Actions: Recovering the Syntax and Semantics of Goal-Directed Human Activities. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 780–787. IEEE, 2014. 3

[10] Hsin-Ying Lee, Jia-Bin Huang, Maneesh Singh, and Ming-Hsuan Yang. Unsupervised Representation Learning by Sorting Sequences. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 667–676. IEEE, 2017. 1

[11] Yang Liu, Qianqian Xu, Peisong Wen, Siran Dai, and Qingming Huang. When the Future Becomes the Past: Taming Temporal Correspondence for Self-supervised Video Representation Learning. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 24033–24044. IEEE, 2025. 2

[12] R.Duncan Luce. The choice axiom after twenty years. *Journal of Mathematical Psychology*, 15(3):215–233, 1977. 1

[13] Ishan Misra, C Lawrence Zitnick, and Martial Hebert. Shuffle and Learn: Unsupervised Learning Using Temporal Order Verification. In *Computer Vision – ECCV 2016*, pages 527–544, Cham, 2016. Springer International Publishing. 1

[14] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning Robust Visual Features without Supervision. *arXiv preprint arXiv:2304.07193*, 2023. 4

[15] R L Plackett. The Analysis of Permutations. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 24(2): 193–202, 1975. 1

[16] Pierre Sermanet, Corey Lynch, Jasmine Hsu, and Sergey Levine. Time-Contrastive Networks: Self-Supervised Learning from Multi-view Observation. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 486–487. IEEE, 2017. 1

[17] Ralf Stauder, Daniel Ostler, Michael Kranzfelder, Sebastian Koller, Hubertus Feußner, and Nassir Navab. The TUM LapChole dataset for the M2CAI 2016 workflow challenge. *arXiv preprint arXiv:1610.09278*, 2016. 3, 5

[18] Andru P. Twinanda, Sherif Shehata, Didier Mutter, Jacques Marescaux, Michel de Mathelin, and Nicolas Padoy. EndoNet: A Deep Architecture for Recognition Tasks on Laparoscopic Videos. *IEEE Transactions on Medical Imaging*, 36(1):86–97, 2017. 1, 2, 3, 5

[19] Limin Wang, Bingkun Huang, Zhiyu Zhao, Zhan Tong, Yinan He, Yi Wang, Yali Wang, and Yu Qiao. VideoMAE V2: Scaling Video Masked Autoencoders with Dual Masking. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14549–14560. IEEE, 2023. 4

[20] Ziyi Wang, Bo Lu, Yonghao Long, Fangxun Zhong, Tak-Hong Cheung, Qi Dou, and Yunhui Liu. AutoLaparo: A New Dataset of Integrated Multi-tasks for Image-guided Surgical Automation in Laparoscopic Hysterectomy. In *Medical Image Computing and Computer Assisted Intervention – MICCAI 2022*, pages 486–496, Cham, 2022. Springer Nature Switzerland. 3, 5, 6

[21] Donglai Wei, Joseph Lim, Andrew Zisserman, and William T Freeman. Learning and Using the Arrow of Time. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8052–8060. IEEE, 2018. 1

[22] Dejing Xu, Jun Xiao, Zhou Zhao, Jian Shao, Di Xie, and Yueting Zhuang. Self-Supervised Spatiotemporal Learning via Video Clip Order Prediction. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10326–10335. IEEE, 2019. 1

[23] Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan Yuille, and Tao Kong. iBOT: Image BERT Pre-Training with Online Tokenizer. *International Conference on Learning Representations (ICLR)*, 2022. 2, 4