LEMON: A Large Endoscopic MONocular Dataset and Foundation Model for Perception in Surgical Settings

Supplementary Material

A. Surgical Background

In contrast to traditional open surgery (not the target of our work), which entails extensive tissue disruption via large incisions and relies on direct visual inspection by the surgeon (i.e., no cameras are used), modern surgical practices predominantly employ minimally invasive techniques (the target of our work). These techniques involve the insertion of slender instruments, including one that is a camera, into the patient's body through small incisions.

To control the instruments from outside the patient's body, there are two variants: robotic-assisted surgery (called "robotic" in our manuscript) and non-robotic laparoscopy (called "non-robotic" in our manuscript). Robotic-assisted surgery involves a surgeon sitting on a console and controlling (with joysticks) several robotic arms that hold and steer the instruments inside the patient. In contrast, non-robotic (also known as traditional or conventional) laparoscopy requires the surgeon to directly hold and steer the laparoscopic hand-held instruments (i.e., no robotic arms involved).

B. Data Curation Details

This section details our data curation pipeline and presents the procedure diversity and distribution of LEMON (Fig. 1). From an initial pool of 18K raw videos, our video classification filtering retained 6617. Subsequent trimming and preprocessing removed 66 hours of non-surgical footage, yielding the final LEMON dataset of 4194 videos (938 hours).

B.1. Video Classification

Video summarization. To improve the efficiency of annotating surgical and non-surgical videos, we obtained 4×4-image video storyboards (i.e., a single image containing a collage of key video frames) for all the collected videos using the method described in [11]. Storyboards enabled us to quickly determine whether a video contained substantial surgical footage, thereby avoiding the need for complex analysis on the entire sequence.

Video storyboard classification. We manually annotated a dataset comprising 2160 surgical and 1910 non-surgical storyboards. The annotation criterion adopted for labeling a storyboard as *surgical* was that at least 50% of the key frames contained shots from a surgical camera; specifically, open surgery videos were categorized as non-surgical. To classify the rest of the collected videos, we trained a ResNet18 [13]. To ensure the accuracy of the inference results, we manually reviewed all the videos classified as surgical.

Table 1. Performance of the video storyboard classification models across five folds.

Fold	F1-score (%)	Accuracy (%)	Precision (%)	Recall (%)
Fold 0	94.42	94.50	95.88	93.00
Fold 1	93.60	93.50	92.23	95.00
Fold 2	93.89	93.75	91.87	96.00
Fold 3	96.50	96.50	96.50	96.50
Fold 4	95.36	95.50	98.40	92.50
Average	94.75	94.75	94.98	94.60
Std Dev	1.10	1.16	2.66	1.57

Performance of video storyboard classification models. We trained five video storyboard classification models (different data splits) to categorize the videos as either surgical or non-surgical. Each fold was split into training, validation, and testing sets with ratios of 0.8, 0.1, and 0.1, respectively. The average F1-score of the video storyboard classification models was $94.75\% \pm 1.1$. The results for each cross-validation fold are shown in Table 1.

B.2. Video Selection and Trimming

Frame classification. A ResNet18 [13] was trained for the surgical/non-surgical video frame classification task. To produce the annotations, the videos were sampled at one frame per second (fps). We annotated 7967 frames, 5481 of which turned out to be surgical and 2486 non-surgical.

Video trimming. Most online videos contain introductory and conclusion slides. Experimentally, we found that the start and end of the surgical footage can be reliably identified by finding the first and last three consecutive frames classified as surgical by our surgical frame classifier (sampling the video at 1 fps). Therefore, this is the approach we used to discard the non-surgical parts at the beginning and end of the collected videos. The resulting videos were manually quality checked.

Performance of frame classification models. We trained five frame classification models to classify a video frame as either surgical or non-surgical. Each model of the five was trained in a different training-validation-testing split of the data, with a split rate of 0.8, 0.1, and 0.1. As shown in Table 2, the average F1-score of the frame classification models was $95.64\% \pm 0.94$.

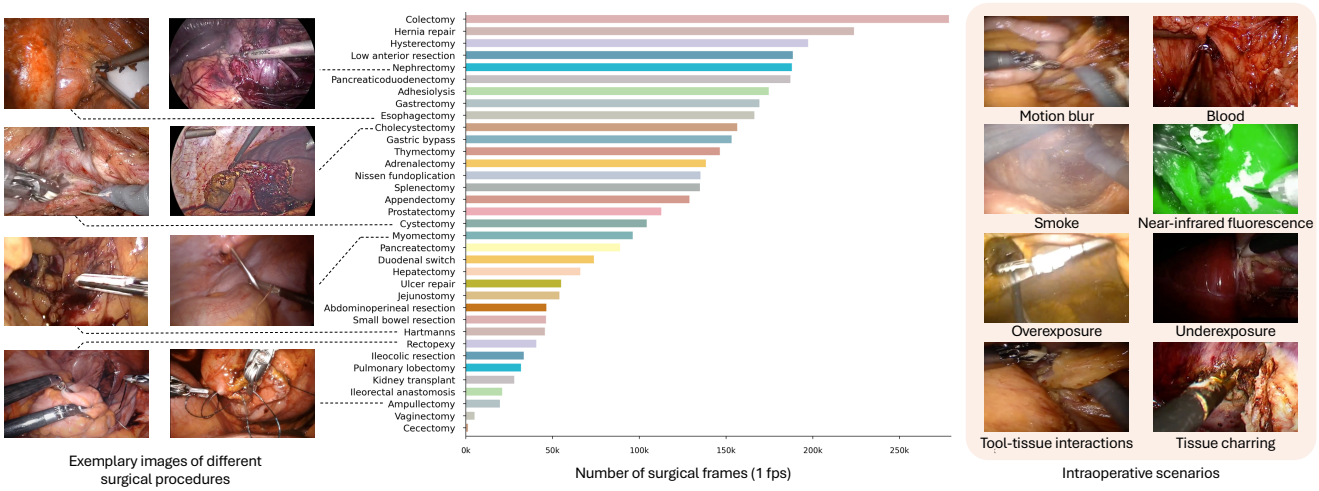

Figure 1. **Diversity and procedure prevalence in LEMON.** Representative samples from various procedures, demonstrating the diverse range of cases in our curated dataset (left, right). Distribution of surgical frames by procedure type (center).

B.3. Video Preprocessing

We use our trained frame classifier to detect and remove the intraoperative non-surgical frames. Additionally, we manually annotated 2719 surgical frames with 4584 non-surgical bounding-box instances and trained a YOLOv8 Nano model [20] to detect and obliterate the non-surgical content in surgical frames. The positions of the non-surgical content bounding boxes that have been obliterated from the video frames in LEMON are provided in a JSON file for those researchers who wish to know where the coordinates of the non-surgical information (e.g., UI elements containing instrument names) are located in the original frames. After this process, we manually quality controlled all the remaining curated videos.

Performance of non-surgical content detection models. We trained five non-surgical content detection models to detect and obliterate *non-surgical* regions in *surgical* video frames. Each model was trained on a different training-validation-testing split of the data, with a split ratio of 0.8, 0.1, and 0.1, respectively. The average mAP50 of the five models was $79.29\% \pm 3.4$, and the average mAP50-95 was $66.18\% \pm 2.98$. The results for all the cross-validation folds are shown in Table 3.

B.4. Video Annotation

For the surgery type, a video is considered to be robotic if the video title includes any of the following keywords: *Robotic*, *Robot*, *Robo*, *Hugo*, *Versius*, *Senhance*, *Telerobotic*, *Console*, and *da Vinci*. The search for these terms in the video titles was case-insensitive. The remaining videos, with titles that do not include any of these keywords, were manually verified to ensure that they are manual surgical procedures. For the surgical procedure type, we cross-reference the video titles with the predefined list of procedures. When exact title matches are not found, we leverage the capabilities of the ChatGPTv4 API to perform a more nuanced analysis, incorporating a customized prompt as shown in Fig. 2. All annotations were manually quality controlled.

Table 2. Performance of frame classification models across five folds.

Fold	F1-score (%)	Accuracy (%)	Precision (%)	Recall (%)
Fold 0	94.65	94.85	98.37	91.21
Fold 1	96.43	96.48	97.93	94.98
Fold 2	96.53	96.61	98.69	94.47
Fold 3	94.44	94.60	97.33	91.71
Fold 4	96.15	96.23	98.17	94.22
Average	95.64	95.75	98.10	93.32
Std Dev	0.94	0.88	0.49	1.39

After curation, the final LEMON dataset consisted of 4194 surgical videos across 35 distinct procedure types, with 94% in 1280×720 resolution and the remaining 6% in varied resolutions, the smallest being 640×480 pixels.

B.5. Curation Precision Strategy

To ensure high precision during dataset curation, we enforced strict filtering criteria. We utilized an ensemble of five models derived from 5-fold cross-validation for our three curation components: video storyboard classification, video frame classification, and non-surgical region detection. During inference, each model applied a confidence threshold of 70%. Final predictions were determined via majority voting (for storyboard and frame classification) and non-maximum suppression (for region detection). This

```
You are a highly knowledgeable assistant specializing in surgical procedures and medical terminology.
Your expertise includes identifying and categorizing surgical interventions based on clinical
    descriptions and procedural contexts.
Here is a list of 35 possible surgical procedure types: pancreatectomy, pancreaticoduodenectomy,
    splenectomy, ampullectomy, hepatectomy, nephrectomy, low anterior resection, colectomy,
    abdominoperineal resection, pulmonary lobectomy, hartmanns, prostatectomy, gastric bypass, duodenal
     switch, gastrectomy, small bowel resection, hernia repair, ulcer repair, cholecystectomy,
    appendectomy, ileocolic resection, cecectomy, myomectomy, hysterectomy, nissen fundoplication,
    adrenalectomy, thymectomy, rectopexy, adhesiolysis, esophagectomy, cystectomy, jejunostomy,
    ileorectal anastomosis, kidney transplant, vaginectomy.
Based on the description of the surgical video: <video title>, determine the most likely procedure type
     from the list. Focus on matching the description to the procedure type that best aligns with the
    terminology and context provided.
```

Figure 2. ChatGPT prompt employed to match video titles to procedure types. The title of the video to be matched is inserted where the <video title> tag is located.

rigorous approach proved highly effective: our final review confirmed that the pipeline achieved 100% precision at the video level and $> 99.9\%$ precision at the frame level.

For the annotation verification, we confirmed that the automated pipeline generated procedure-type labels (e.g., thymectomy, cystectomy) with 95.2% accuracy, and surgery-type labels (e.g., robotic, non-robotic) with 97.6% accuracy. The experts subsequently corrected all identified errors.

B.6. Human Effort

We provide a detailed breakdown of manual effort (in person-hours). The curation workload comprises labeling 4K storyboards (3 h) and 8K frames (3 h), followed by annotating 2719 surgical frames with 4584 non-surgical bounding boxes (10 h). The most extensive process involved manually reviewing LEMON videos, which were previously curated by models trained on the initial annotations, to confirm surgical content and remove artifacts such as out-of-body views (72 h), and finally verifying video annotations for surgery and procedure types (18 h).

B.7. Data Quality and Reliability

To validate LEMON's labels, two researchers (Chengan Che and Chao Wang) independently annotated a stratified random subset of **500 videos (>10%)**, balanced across 35 procedure types. We achieved a **Cohen's Kappa** of **0.97** for procedure types and **>0.99** for surgery types, confirming robust label reliability. This high agreement reflects the clear visual distinctiveness of the procedures and the rich original metadata provided by verified medical professionals.

C. LemonFM Pretraining

C.1. Pretraining Details

For LemonFM pretraining, we trained on an Ubuntu 22.04.5 LTS node with eight NVIDIA V100 GPUs (32GB each), using a batch size of 24 per GPU under PyTorch 2.5.1+cu124 (CUDA 12.4). We employed AdamW as the optimizer with a teacher temperature of 0.04, fp16 precision, an initial learning rate of 5e-4 (after warm-up), a minimum learning rate of 1e-6, and a random seed of 30. The model was trained for 60 epochs, including 10 warm-up epochs. This number of epochs enables the model to converge and stabilize its training loss on our dataset and pretext task. The model with the lowest training loss was selected as the final model.

C.2. Augmented Distillation Design Choices

Cosine Similarity vs. L2 distance. We use cosine similarity for neighbor mining, following standard SSL practice (e.g., SwAV [4], NNCLR [9]). We validate this choice against L2 (Euclidean) distance using a k-NN evaluation for phase recognition, where labels are predicted by a majority vote over the 20 nearest training-set neighbors. To isolate the impact of the distance metric, these experiments were conducted using a ConvNeXt-L [22] backbone pretrained with the vanilla DINO method [5], excluding our proposed surgical augmentations. Results show that cosine similarity significantly outperforms L2 distance, yielding accuracy gains of 4 pp (71.3 to 75.3) on AutoLaparo and 3.5 pp (66.8 to 70.3) on Cholec80.

Sensitivity to augmented distillation threshold. We further analyze the impact of the cosine distance threshold for our augmented distillation (Fig. 4 in the main manuscript). To perform this assessment in a computationally efficient manner, we employ a ConvNeXt-S backbone. Linear probing on AutoLaparo phase recognition yields Acc/F1 scores of 72.9/63.2 ($1.5\times$), **73.4/63.7** ($3\times$), and 72.6/63.0 ($6\times$). Similarly, Cholec80 phase recognition yields Acc/F1 scores of 72.2/65.2 ($1.5\times$), **73.6/66.7** ($3\times$), and 72.7/65.2 ($6\times$). These results confirm that our $3\times$ threshold is near-optimal. Qualitatively, a stricter $1.5\times$ threshold selects too few cross-video neighbors (reducing diversity), while a looser $6\times$ threshold admits dissimilar, noisy views that slightly de-

Table 3. Performance of bounding box detection models across five folds.

Fold	Images	Instances	Precision	Recall	mAP50	mAP50-95
Fold 0	272	447	74.79	77.63	75.50	62.00
Fold 1	272	438	70.73	77.40	76.36	61.79
Fold 2	272	509	83.34	79.60	84.89	68.56
Fold 3	272	422	75.27	83.69	80.31	68.98
Fold 4	272	496	81.32	81.45	82.58	68.65
Average	–	–	77.09	79.94	79.29	66.18
Std Dev	–	–	4.41	2.33	3.40	2.98

grade performance.

D. Downstream Task Details

In this section, we provide details on the datasets used for evaluating each downstream task, including their respective data splits, as well as training configurations for linear probing and full fine-tuning settings. The experiments were conducted on an Ubuntu 22.04.5 LTS node equipped with an Intel IceLake Xeon CPU (72 vCPUs) and two NVIDIA A100 GPUs (each with 80 GB of memory), using PyTorch 2.5.1+cu124 (CUDA 12.4).

D.1. Data Splits

Surgical phase recognition. For AutoLaparo, we followed the standard split of ten training videos, four for validation, and seven for testing [29]. For M2CAI16, we followed the data splits in [15, 19, 28], dividing the dataset into 27 training videos and 14 testing videos, respectively. For Cholec80, we adopted the data splits in [7, 16, 21, 28], allocating 40 videos for training, eight for validation, and 32 for testing.

Surgical tool presence detection. For GraSP, we adopted the data splits specified in [2], with four videos for training, four for validation, and five for testing. The data split for Cholec80 is identical to those employed in the surgical phase recognition task.

Surgical action recognition. For CholecT50, we followed the formal data split as proposed in [25].

Surgical semantic segmentation. For CholecSeg8k, we followed previous works [12, 14] and used 75% of the videos for training and 25% of the videos for testing (videos 12, 20, 48 and 55).

D.2. Linear Probing

We used the teacher model from LemonFM as our backbone. The images were resized to 224×224 for all downstream tasks. The downstream models with linear heads were trained with a batch size of 512, an initial learning rate (LR) of 1e-3, the AdamW optimizer, a random seed of 30, and cross-entropy loss. An early stopping criterion was applied, stopping training after 10 epochs if the validation loss shows no improvement.

Table 4. Video-level linear probing results. Performance of LemonFM is compared against SurgeNetXL across three surgical datasets. Metrics are reported as Accuracy/F1-score.

Method	AutoLaparo	Cholec80	M2CAI16
SurgeNetXL [14]	74.0/50.7	73.4/51.9	64.5/42.9
LemonFM	**76.9/53.8**	**75.1/52.7**	**66.7/44.1**

Furthermore, we report video-level linear probing results to compare LemonFM against SurgeNetXL, a leading surgical foundation model (Table 4).

D.3. Full Fine-tuning

We used the teacher model from LemonFM as our backbone. The images were resized to 224×224 for all downstream tasks. The downstream models were trained with a batch size of 112, an initial learning rate (LR) of 1e-4, the AdamW optimizer, a random seed of 30, and cross-entropy loss. We used five-fold cross-validation for 50% shot fine-tuning. An early stopping criterion was applied, stopping training after 10 epochs if the validation loss shows no improvement.

Surgical phase recognition. To compare with other specialist models that are specifically tailored for this task, we adopt the video-level accuracy and Jaccard in the full fine-tuning setting. The accuracy for a video is computed by dividing the number of frames whose class has been correctly predicted by the total number of frames. The overall accuracy for a dataset is defined as the mean accuracy value over all videos, henceforth referred to as *video-level* accuracy [16–18, 21]. The Jaccard is computed for each class and video, then averaged: first within videos, and secondly across classes [16–18, 21].

During training, we utilized a TCN [7] head and followed the two-stage training approach outlined in TeCNO [7]. In the first stage, we trained the backbone with a linear head to perform frame-wise classification without temporal context. Then, we employed the TCN head to incorporate temporal information of the extracted features for predictions. The default TCN configuration was used, consisting of two stages, each containing nine layers. The TCN head was trained using the Adam optimizer with an initial learning rate of 5e-3 for 70 epochs to ensure sufficient training. The checkpoint with the lowest validation loss was selected as the optimal checkpoint.

D.4. Computational Cost Comparison

We analyze the computational efficiency of our method compared to previous state-of-the-art baselines in Table 5.

Table 5. Computational cost analysis. Inference and training times are measured in milliseconds (ms). Memory usage is reported in Megabytes (MB).

Model	Inference		Training		Acc (%)
	Time	GPU	Time	GPU	
Endo-FM [30]	6.9	221	50.4	891	51.5
EndoViT [3]	6.7	215	47.8	874	45.4
SurgeNet [14]	12.7	157	61.7	941	68.8
GSViT [26]	22.4	56	105.8	178	22.0
LemonFM (CN-B)	7.8	194	54.8	1068	74.7
LemonFM (CN-L)	8.9	387	58.7	1782	76.4

To support diverse computational constraints, we provide LemonFM with two backbone variants: ConvNeXt-Base and ConvNeXt-Large, both of which will be publicly released.

All measurements were conducted on an NVIDIA A100 GPU using 224×224 images in FP16 precision. Metrics include inference time (per image), training time (per step with batch size 8), peak GPU memory usage, and linear probing accuracy on AutoLaparo phase recognition. While strictly lightweight baselines like EndoViT achieve the lowest inference latency (6.7 ms), they suffer from limited representational capacity (45.4% accuracy). In contrast, our LemonFM (CN-Base) maintains highly competitive efficiency (7.8 ms inference) while delivering a substantial performance leap of nearly 30 percentage points (74.7%). Even our larger variant (CN-Large) remains efficient (8.9 ms) while further pushing accuracy to 76.4%. This demonstrates that LemonFM achieves significant performance gains while maintaining practical computational efficiency suitable for real-time applications.

E. Evaluation of LemonFM-Vid

To evaluate the performance of our video classifier, LemonFM-Vid, we split the LEMON dataset by videos, with 3369 videos for training and 825 videos for testing. Our vanilla LemonFM was trained on all the LEMON frames; therefore, to evaluate the video classification performance fairly, we trained a new LemonFM model from scratch using only the frames of the 3369 videos in the training set that are accessible to all other classifiers in the comparison.

F. Dataset and Licensing

We adhere to the same practices as other datasets created from YouTube and various sources on the Web, such as ImageNet [8], Kinetics [6], YouTube-VIS [32], YouTube-8M [1], Insect-1M [24], Moments-in-Time [23], Tai-Chi-HD [27], HD-Villa-100M [31], and AVSpeech [10]. Specifically, we publicly release the dataset metadata, the list of original YouTube video IDs, and our corresponding annotations under the Creative Commons Attribution 4.0 International (CC BY 4.0) license. The copyright of the raw videos remains with their original creators.

To facilitate reproducibility, verified researchers at academic institutions can directly request access to the curated LEMON video files for non-commercial research purposes, subject to a strict data access agreement. Furthermore, we provide an online form for original YouTube video authors to explicitly opt out and request the removal of their content from our dataset.

References

[1] Sami Abu-El-Haija, Nisarg Kothari, Joonseok Lee, Paul Natsev, George Toderici, Balakrishnan Varadarajan, and Sudheendra Vijayanarasimhan. YouTube-8M: A Large-Scale Video Classification Benchmark. *arXiv preprint arXiv:1609.08675*, 2016. 5

[2] Nicolás Ayobi, Santiago Rodríguez, Alejandra Pérez, Isabela Hernández, Nicolás Aparicio, Eugénie Dessevres, Sebastián Peña, Jessica Santander, Juan Ignacio Caicedo, Nicolás Fernández, and Pablo Arbeláez. Pixel-wise recognition for holistic surgical scene understanding. *Medical Image Analysis*, 106:103726, 2025. 4

[3] Dominik Batić, Felix Holm, Ege Özsoy, Tobias Czempiel, and Nassir Navab. EndoViT: pretraining vision transformers on a large collection of endoscopic images. *International Journal of Computer Assisted Radiology and Surgery*, 19(6): 1085–1091, 2024. 5

[4] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, Red Hook, NY, USA, 2020. Curran Associates Inc. 3

[5] 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. 3

[6] 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. 5

[7] Tobias Czempiel, Magdalini Paschali, Matthias Keicher, Walter Simson, Hubertus Feussner, Seong Tae Kim, and Nassir Navab. TeCNO: Surgical Phase Recognition with Multi-stage Temporal Convolutional Networks. In *Medical Image Computing and Computer Assisted Intervention – MICCAI 2020*, pages 343–352, Cham, 2020. Springer International Publishing. 4

[8] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255. IEEE, 2009. 5

[9] Debidatta Dwibedi, Yusuf Aytar, Jonathan Tompson, Pierre Sermanet, and Andrew Zisserman. With a Little Help from My Friends: Nearest-Neighbor Contrastive Learning of Visual Representations. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9568–9577. IEEE, 2021. 3

[10] Ariel Ephrat, Inbar Mosseri, Oran Lang, Tali Dekel, Kevin Wilson, Avinatan Hassidim, William T Freeman, and Michael Rubinstein. Looking to listen at the cocktail party: a speaker-independent audio-visual model for speech separation. *ACM Trans. Graph.*, 37(4), 2018. 5

[11] Luis C. Garcia-Peraza-Herrera, Sebastien Ourselin, and Tom Vercauteren. VideoSum: A Python Library for Surgical Video Summarization. In *Conference on New Technologies for Computer and Robot Assisted Surgery*, 2023. 1

[12] Maria Grammatikopoulou, Ricardo Sanchez-Matilla, Felix Bragman, David Owen, Lucy Culshaw, Karen Kerr, Danail Stoyanov, and Imanol Luengo. A spatio-temporal network for video semantic segmentation in surgical videos. *International Journal of Computer Assisted Radiology and Surgery*, 19(2):375–382, 2023. 4

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778. IEEE, 2016. 1

[14] Tim J.M. Jaspers, Ronald L.P.D. de Jong, Yiping Li, Carolus H.J. Kusters, Franciscus H.A. Bakker, Romy C. van Jaarsveld, Gino M. Kuiper, Richard van Hillegersberg, Jelle P. Ruurda, Willem M. Brinkman, Josien P.W. Pluim, Peter H.N. de With, Marcel Breeuwer, Yasmina Al Khalil, and Fons van der Sommen. Scaling up self-supervised learning for improved surgical foundation models. *Medical Image Analysis*, 108:103873, 2026. 4, 5

[15] Yueming Jin, Qi Dou, Hao Chen, Lequan Yu, Jing Qin, Chi-Wing Fu, and Pheng-Ann Heng. SV-RCNet: Workflow Recognition From Surgical Videos Using Recurrent Convolutional Network. *IEEE Transactions on Medical Imaging*, 37(5):1114–1126, 2018. 4

[16] Yueming Jin, Huaxia Li, Qi Dou, Hao Chen, Jing Qin, Chi-Wing Fu, and Pheng-Ann Heng. Multi-task recurrent convolutional network with correlation loss for surgical video analysis. *Medical Image Analysis*, 59:101572, 2020. 4

[17] Yueming Jin, Yonghao Long, Cheng Chen, Zixu Zhao, Qi Dou, and Pheng-Ann Heng. Evaluation code from Temporal Memory Relation Network for Workflow Recognition From Surgical Video, 2021.

[18] Yueming Jin, Yonghao Long, Cheng Chen, Zixu Zhao, Qi Dou, and Pheng-Ann Heng. Temporal Memory Relation Network for Workflow Recognition From Surgical Video. *IEEE Transactions on Medical Imaging*, 40(7):1911–1923, 2021. 4

[19] Yueming Jin, Yonghao Long, Xiaojie Gao, Danail Stoyanov, Qi Dou, and Pheng-Ann Heng. Trans-SVNet: hybrid embedding aggregation Transformer for surgical workflow analysis. *International Journal of Computer Assisted Radiology and Surgery*, 17(12):2193–2202, 2022. 4

[20] Glenn Jocher, Ayush Chaurasia, and Jing Qiu. Ultralytics YOLO, 2023. 2

[21] Yang Liu, Maxence Boels, Luis C. Garcia-Peraza-Herrera, Tom Vercauteren, Prokar Dasgupta, Alejandro Granados, and Sébastien Ourselin. LoViT: Long Video Transformer for surgical phase recognition. *Medical Image Analysis*, 99: 103366, 2025. 4

[22] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A ConvNet for the 2020s. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11966–11976. IEEE, 2022. 3

[23] Mathew Monfort, Carl Vondrick, Aude Oliva, Alex Andonian, Bolei Zhou, Kandan Ramakrishnan, Sarah Adel Bargal, Tom Yan, Lisa Brown, Quanfu Fan, and Dan Gutfreund. Moments in Time Dataset: One Million Videos for Event Understanding. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(2):502–508, 2020. 5

[24] Hoang-Quan Nguyen, Thanh-Dat Truong, Xuan Bac Nguyen, Ashley Dowling, Xin Li, and Khoa Luu. Insect-Foundation: A Foundation Model and Large-Scale 1M Dataset for Visual Insect Understanding. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 21945–21955. IEEE, 2024. 5

[25] Chinedu Innocent Nwoye, Tong Yu, Cristians Gonzalez, Barbara Seeliger, Pietro Mascagni, Didier Mutter, Jacques Marescaux, and Nicolas Padoy. Rendezvous: Attention mechanisms for the recognition of surgical action triplets in endoscopic videos. *Medical Image Analysis*, 78:102433, 2022. 4

[26] Samuel Schmidgall, Ji Woong Kim, Jeffrey Jopling, and Axel Krieger. General surgery vision transformer: A video pre-trained foundation model for general surgery. *arXiv preprint arXiv:2403.05949*, 2024. 5

[27] Aliaksandr Siarohin, Stéphane Lathuilière, Sergey Tulyakov, Elisa Ricci, and Nicu Sebe. First order motion model for image animation. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*. Curran Associates Inc., Red Hook, NY, USA, 2019. 5

[28] 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. 4

[29] 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. 4

[30] Zhao Wang, Chang Liu, Shaoting Zhang, and Qi Dou. Foundation Model for Endoscopy Video Analysis via Large-Scale Self-supervised Pre-train. In *Medical Image Computing and Computer Assisted Intervention – MICCAI 2023*, pages 101–111, Cham, 2023. Springer Nature Switzerland. 5

[31] Hongwei Xue, Tiankai Hang, Yanhong Zeng, Yuchong Sun, Bei Liu, Huan Yang, Jianlong Fu, and Baining Guo. Advancing High-Resolution Video-Language Representation

with Large-Scale Video Transcriptions. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5026–5035. IEEE, 2022. 5

[32] Linjie Yang, Yuchen Fan, and Ning Xu. Video Instance Segmentation. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 5187–5196. IEEE, 2019. 5