scientific **data**

DATA DESCRIPTOR

Check for updates

CholecInstanceSeg: A Tool Instance Segmentation Dataset for Laparoscopic Surgery

Oluwatosin Alabi[1], Ko Ko Zayar Toe[2], Zijian Zhou[3], Charlie Budd[1], Nicholas Raison[1], Miaojing Shi[4] ✉ & Tom Vercauteren[1]

In laparoscopic and robotic surgery, precise tool instance segmentation is an essential technology for advanced computer-assisted interventions. Although publicly available procedures of routine surgeries exist, they often lack comprehensive annotations for tool instance segmentation. Additionally, the majority of standard datasets for tool segmentation are derived from porcine(pig) surgeries. To address this gap, we introduce CholecInstanceSeg, the largest open-access tool instance segmentation dataset to date. Derived from the existing CholecT50 and Cholec80 datasets, CholecInstanceSeg provides novel annotations for laparoscopic cholecystectomy procedures in patients. Our dataset comprises 41.9k annotated frames extracted from 85 clinical procedures and 64.4k tool instances, each labelled with semantic masks and instance IDs. To ensure the reliability of our annotations, we perform extensive quality control, conduct label agreement statistics, and benchmark the segmentation results with various instance segmentation baselines. CholecInstanceSeg aims to advance the field by offering a comprehensive and high-quality open-access dataset for the development and evaluation of tool instance segmentation algorithms.

Background & Summary

Minimally invasive surgeries (MIS) have gained popularity due to their benefits over open surgeries, such as reduced blood loss, less pain, faster recovery times, and fewer post-surgical complications[1]. However, MIS relies on an endoscope or laparoscope for indirect vision, which provides a limited field of view, making it challenging for surgeons to accurately interpret complex surgical scenes[2]. To overcome this, computer-aided intervention techniques that enhance the information obtained through minimally invasive cameras have been proposed[3,4]. At the forefront of this evolution lies the challenge of achieving precise instance segmentation of surgical tools in a minimally invasive surgical scene. This task is essential for developing assistive surgical technology and autonomous or semi-autonomous surgical systems, which require detailed knowledge of the identification and localization of various tools.

Current publicly available datasets for surgical tool segmentation have several notable limitations. Many focus on porcine or *ex-vivo* surgeries rather than routine human procedures, limiting their applicability to day-to-day clinical settings[5–7]. Others are multitask datasets with limited tool segmentation data[8], or they provide only binary instance segmentation for a single tool class[9]. Additionally, available surgical tool segmentation datasets lack instance-specific information[5–8,10,11], and most offer a relatively small number of annotated segmentation masks.

To overcome these challenges, we introduce the CholecInstanceSeg open-access dataset, in which we meticulously annotate 41,933 frames from the existing Cholec80[12] and CholecT50[13] datasets with instance segmentation labels and class labels for 7 instrument classes: Grasper, Bipolar, Hook, Clipper, Scissors, Irrigator, and Snare. CholecInstanceSeg contains frames from 85 videos of Laparoscopic cholecystectomy - a routine surgical procedure for gallbladder removal, commonly performed to treat gallstones or cholecystitis with over 300,000 surgeries performed annually in the United States[14]. To the best of our knowledge, CholecInstanceSeg represents the most extensive surgical tool instance segmentation dataset. Table 1 compares popular datasets used in

[1]Kings College London, Surgical & Interventional Engineering, London, SE1 7EU, United Kingdom. [2]Kings College Hospital Denmark Hill, department, London, SE5 9RS, United Kingdom. [3]Department of Informatics, King's College London, London, United Kingdom. [4]Tongji University, College of Electronic and Information Engineering, Shanghai, 200092, China. ✉e-mail: mshi@tongji.edu.cn

Dataset	Patient	Environment	Procedures	Annotations	Semantic	Instance
Endovis2015[5]	Human	*Ex-vivo* and *In-vivo*	8	10k	3 classes	No
Endovis2017[6]	Porcine	*In-vivo*	10	2.4k	10 classes	No
Endovis2018[7]	Porcine	*In-vivo*	16	2.4k	10 classes	No
CholecSeg8k[11]	Human	*In-vivo*	17	8k	12 classes	No
AutoLaparo (Multitask)[8]	Human	*In-vivo*	21	1.8k	7 classes	No
ROBUSTMIS2019[9]	Human	*In-vivo*	30	10k	2 classes	Yes✓
SAR-RARP50 (Multitask)[10]	Human	*In-vivo*	50	10k	10 classes	No
CholecInstanceSeg (Ours)	Human	*In-vivo*	85	41.9k	8 classes	Yes✓

Table 1. Comparison of popular datasets used in tool segmentation research, indicating the type of surgeries, the number of surgical procedures, the number of frames annotated, the semantic classes annotated, and whether each dataset contains instance information.
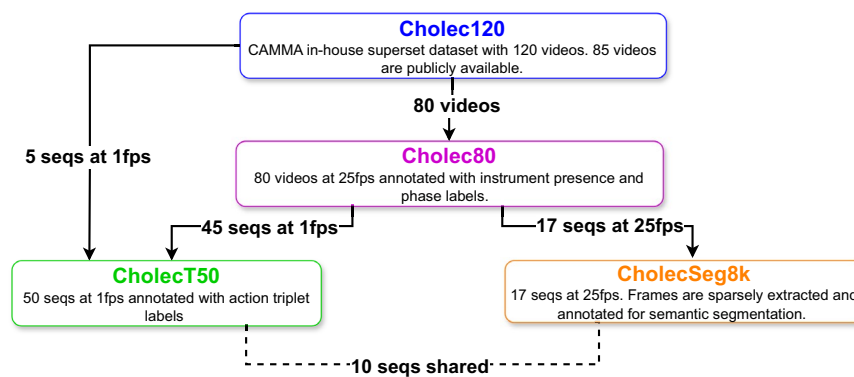

Fig. 1 CholecInstanceSeg contains frames extracted from CholecT50, CholecSeg8k, and Cholec80. These three datasets are sub-datasets of the CAMMA in-house dataset called Cholec120. There are also some shared image sequences (referred to as *seqs* in the figure for brevity) between CholecT50 and CholecSeg8k.

tool segmentation research and CholecInstanceSeg. We acknowledge a concurrent effort, the PhaKIR dataset, which is being developed for the upcoming MICCAI2024 PhaKIR-challenge. The PhaKIR dataset is expected to include 15 videos and approximately 30,000 instance annotations. However, as this dataset is only partially released and not yet available for research, we have not included it in our comparisons.

Furthermore, we provide technical validations of our dataset, demonstrating our comprehensive quality control procedures and showcasing CholecInstanceSeg's capability for training instance segmentation models.

CholecInstanceSeg is ideally suited for training models on instance and semantic segmentation for laparoscopic surgeries. Additionally, it can be effectively integrated with other datasets that utilize surgical procedure videos from Cholec80 and CholecT50. These datasets, derived from Cholec80 and CholecT50 share frames with CholecInstanceSeg but are annotated for different tasks, such as phase recognition (Cholec80[12]) instrument action and target detection (CholecT50[13]), and Critical View of Safety (CVS) assessment (Cholec80-CVS[15]).

Methods

This section outlines the methodology for creating the CholecInstanceSeg dataset, divided into four main topics: data sources, labelling protocol, annotation tool and techniques, and the annotation workflow and procedures.

First, we describe the data sources and how the frames were extracted. Next, we outline the principles guiding our instance segmentation dataset, emphasizing the labelling protocol. We then discuss the tools and techniques used to overcome annotation challenges. Finally, we provide a detailed account of the annotation process, including information about the annotators and the time spent on creating CholecInstanceSeg.

Data Sources.
The CholecInstanceSeg dataset is curated using frames extracted from three existing laparoscopic cholecystectomy datasets: Cholec80[12], CholecT50[13], and CholecSeg8k[11], containing a total of 85 image sequences. These datasets are publicly available subsets of the in-house Cholec120 dataset[16] from the CAMMA research group which was recorded at the University Hospital of Strasbourg[12,13]. The interrelation between these datasets is illustrated in Fig. 1.

When discussing the source datasets, we distinguish between videos and image sequences. Videos are multimedia compressed files provided in formats such as.mp4, while image sequences refer to extracted sequential frames from videos.

The frames from Cholec80 and CholecT50 used in CholecInstanceSeg are primarily at a resolution of 854×480 pixels, consistent with the original Cholec80 and CholecT50 datasets. However, three sequences from CholecT50 are provided at a higher resolution of 1920×1080 pixels. CholecT50 contains pre-extracted image sequences and we used these as is, while frames from Cholec80 were extracted at 1 fps using the FFMPEG library.

CholecInstanceSeg includes frames from publicly available datasets that already have the necessary consents and ethical considerations. Additionally, the licenses for the source datasets permit the enhancement of the data with further annotations (Cholec80 license, CholecT50 license, CholecSeg8k license).

CholecSeg8k. CholecSeg8k comprises 8,080 laparoscopic cholecystectomy image frames extracted from 17 surgical procedure video clips within the Cholec80 dataset. This dataset provides 80 consecutive images sparsely annotated for semantic segmentation (but not instance segmentation). To enhance CholecSeg8k, instance IDs are required for the existing semantic segmentation labels. As CholecSeg8k already contains semantic segmentation labels, it serves as a good starting point for further annotation. However, it is important to note that CholecSeg8k includes only two tool classes: Grasper and Hook, which is fewer than the number of tool classes in the parent datasets.

CholecT50. CholecT50 is a publicly available dataset comprising 50 real-world videos of laparoscopic chole-cystectomy surgeries. This dataset integrates 45 videos from the Cholec80 dataset along with 5 additional videos from the in-house Cholec120 dataset. The image sequences in CholecT50 are extracted at a rate of 1 frame per second (fps). Each frame in CholecT50 is annotated with fine-grained tool-tissue interaction labels, specifying the `<instrument, action, target>`. Although CholecSeg8k and CholecT50 share 10 image sequences, we confirmed that the video extraction methods used for these datasets are different. We checked the similarity between extracted frames in the shared sequences from both datasets and found no exact matches.

The CholecT50 dataset consists of 100.9k frames distributed across 50 image sequences. Given the high cost of annotating every single frame, we adopted a strategy to balance thoroughness and feasibility. Consequently, we divided CholecT50 into two partitions:

1. CholecT50-full: This partition was chosen to feature comprehensive annotations, including instance IDs and segmentation masks for every frame in the image sequence. This thorough annotation provides detailed data for in-depth analysis and model training, allowing for the exploration of ideas like video segmentation.
2. CholecT50-sparse: In this partition, annotations are provided sparsely in time, yet they cover the entire sequence. This approach reduces annotation costs while still offering valuable insights into the location and identification of tools for each sequence. We sample one in every 30 frames ($\frac{1}{30}$ fps) and add more frames if the resulting sample contains fewer than 50 frames per sequence.

Cholec80. Cholec80 comprises 80 videos of cholecystectomy surgeries. It is labelled with phase information at 25 fps and tool presence annotations at 1 fps. The tool annotations include instruments where at least half of the tool-tip is visible. There are six tools available in the provided tool presence annotations.

Given the overlap between Cholec80, CholecSeg8k, and CholecT50, we focus on annotating the 28 videos in Cholec80 that are not present in CholecSeg8k and CholecT50. We extract frames from these 28 videos at a rate of $\frac{1}{30}$ fps and provide annotations for these frames. To maintain consistency with the extraction protocol of CholecT50 and CholecSeg8k, we blacked out frames that showed the exterior of the body via the laparoscope. Additionally, if the resulting sample contained fewer than 50 frames per sequence, more frames were included to ensure adequate coverage.

Summary of CholecInstanceSeg Partitions. CholecInstanceSeg, which contains 41.9k frames from 85 unique image sequences, can be partitioned into four distinct sections based on the data source: Instance-CholecSeg8k, Instance-CholecT50-full, Instance-CholecT50-sparse, and Instance-Cholec80-sparse. Our process for selecting the frames and sequences for each partition is as follows:

1. Instance-CholecSeg8k: We selected all the 8,080 frames from all the 17 sequences provided in CholecSeg8k, we utilized all the images in CholecSeg8k.
2. Instance-CholecT50-full: We randomly selected 15 sequences within CholecT50 (CholecT50-full), we utilized all frames provided by CholecT50 for these 15 sequences - 28,317 frames.
3. Instance-CholecT50-sparse: For the remaining 35(50-15) sequences in CholecT50(CholecT50-sparse), we sampled these sequences by selecting one frame out of every 30 frames ($\frac{1}{30}$ fps), ensuring a minimum of 50 frames in each sequence - 2,681 frames.
4. Instance-Cholec80-sparse: After selecting sequences for Instance-CholecSeg8k, Instance-CholecT50 (both full and sparse), there remained 28 videos from Cholec80, which are not included in CholecT50 and CholecSeg8k, and these videos were sampled at $\frac{1}{30}$ fps, ensuring a minimum of 50 frames in each sequence - 2,855 frames.

A visual representation of dataset partitions in CholecInstanceSeg can be seen in Fig. 2.

The differences in frame rates for each dataset partition reflect both the design of the parent datasets – Cholec80 provides videos at 25fps which we then extract at 1fps and CholecT50 provides image sequences at 1fps directly – and practical considerations for annotating instance segmentation on a large number of frames. Annotating every sequence densely at 1fps is prohibitively expensive. Thus, we adopted a mixed approach, converting existing semantic segmentation to instance segmentation, annotating some sequences *fully* (using all frames available from the parent datasets), and others sparsely (sampling frames at reduced frame rates of $\frac{1}{30}$ fps). This approach allows us to balance annotation density and dataset diversity, ensuring coverage of both densely annotated sequences and a wider range of surgical scenes.
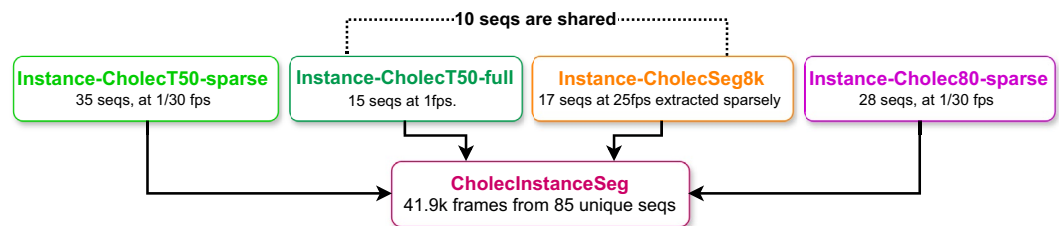
Fig. 2 CholecInstanceSeg is composed of frames from four dataset partitions: Instance-CholecT50-sparse, Instance-CholecT50-full, Instance-CholecSeg8k, and Instance-Cholec80-sparse. The diagram illustrates the number of image sequences (denoted *seqs* in the figure for brevity) and extraction rates for each partition. Note that 10 sequences are shared between Instance-CholecT50-full and Instance-CholecSeg8k.

Figure 7 presents the exact sequences in each partition, represented as VID-XX, where XX is the sequence number.

The Labelling Protocol. To facilitate the creation of a high-quality instance segmentation dataset for laparoscopic cholecystectomy videos, we established specific rules to ensure accurate and consistent identification and segmentation of surgical instruments. These rules are discussed under three main topics: Tool Categories, Annotation Scope, and Annotation Guidelines.

Tool Categories. In this section, we detail the tool categories selected for annotation and discuss the objects that were deliberately excluded, along with the rationale for these decisions.

We identified seven tool categories for annotation based on our observations from the extracted CholecInstanceSeg sequences:

1. Grasper: Standard grasper used for holding tissues.
2. Bipolar: Bipolar grasper used for grasping and coagulation.
3. Hook: Monopolar hook used for dissection and coagulation.
4. Clipper: Clip applier used for applying clips.
5. Scissors: Monopolar scissors used for cutting tissues.
6. Irrigator: Suction/irrigation device used for fluid management.
7. Snare: Laparoscopic snare used for ligation of tissues.

The inclusion of the snare category deviates from the standard six categories (grasper, bipolar, hook, clipper, scissors, irrigator) defined by Cholec80 and CholecT50. We determined that the snare's distinct visual characteristics and unique functionality warranted its classification as a separate tool category.

Annotation Scope. The annotation scope defines the entities included and excluded to maintain a focused and high-quality dataset. While we annotated all laparoscopic tools, some objects used in conjunction with these tools were excluded to streamline the annotation process:

- Excluded Entities: Individual clips from the clipper, surgical needles, surgical sutures, the specimen bag, the camera port, and gauze.
- Instrument Ports: We did not annotate the instrument port when no instrument extended from it. However, when an instrument was seen extending from the port, we annotated the instrument port as part of the instrument. This approach aligns with the CholecSeg8k annotation style and addresses the difficulty of distinguishing the exact point where the port ends and the instrument begins.

Annotation Guidelines. Our annotation guidelines ensure consistency and accuracy in the labelling process. These guidelines cover various scenarios and provide specific instructions for annotators:

- Annotation Quality: We aimed to annotate every visible instrument while maintaining high segmentation quality. Although achieving pixel-perfect annotation for every instance was impractical due to time and budget constraints, we allowed minor imperfections in instrument boundaries, provided that the fundamental characteristics of the instrument and its end-effectors were accurately outlined. Holes within instruments were considered integral parts of the instrument itself, following the standard annotation conventions in popular tool segmentation datasets like EndoVis2017[6] and EndoVis2018[7].
- Annotation of Hard Cases: CholecInstanceSeg captures routine surgical procedures, numerous challenging scenarios were encountered. We accounted for twelve specific situations: motion blur, presence of smoke, occlusion by soft transparent tissues or blood, tissues attached to instruments, instruments at the edge of the frame, instruments far from the camera, light reflection, low lighting conditions, bright lighting conditions, lens dirtiness, camera in camera port, and instruments in liquid. Specialized instructions and examples were provided to annotators, including checking neighbouring frames for clarification, utilizing the expected shape of the instrument when in doubt, annotating as much of the instrument as possible, and annotating even under low visibility conditions.

Annotation Tool and Techniques.　　To provide annotations for CholecInstanceSeg, we needed an annotation tool for performing manual annotations, a method to convert existing semantic segmentation to instance segmentation, and a way to efficiently label a large number of images. This section discusses how we addressed these needs: the annotation tool, converting semantic segmentation to instance segmentation, and semi-automatic annotation.

Annotation Tool.　　Data labelling was primarily executed using a minimally customized version of the Segment Anything Annotator tool[17], which is open-source software based on the LabelMe annotation project[18]. This tool integrates instrument segmentation functionalities and interactive segmentation capabilities, leveraging the Segment Anything Model (SAM)[19]. Our customizations included enhancements for navigation, reduction in the size of annotation label files, and the addition of a quality control mode. The customized version is available on github.

We predominantly utilized point-based SAM prompts for efficient and quick interactive annotation. However, point-prompt-based interactive segmentation often encounters challenges in complex scenarios, such as when instruments and surrounding backgrounds are similar in color, low lighting, glare, blurs, dirty lenses, reflections, and the presence of smoke. In such instances, manual annotation can also be performed using the same tool. Further discussion of challenging scenarios is provided in a later section.

Converting Semantic to Instance Segmentation.　　One of our data sources, CholecSeg8k, already had semantic segmentation labels. However, we aimed to assign instance IDs to distinguish each instrument instance within an image. To add instance labels and convert existing semantic segmentation labels to instance segmentation, we followed two main steps:

1. **Class-Aware Connected Component Analysis**: We employed connected component analysis for each class separately to identify and segregate different instances within the frames. This technique effectively assigned instance IDs to instruments from different classes and correctly distinguished instances of instruments belonging to the same class that do not overlap in pixel space.
2. **Identifying and Addressing Failure Cases**: Class-aware connected component analysis encountered several challenges. A total of 892 frames with these issues were carefully identified and documented. To rectify these problems, we developed and applied several pipelines:

   a. **Occlusions**: When pixels of the same instrument are not connected, class-aware connected component analysis assigns multiple instance IDs to a single instrument. This issue, mainly caused by occlusions at the frame's edge or by tissues, was addressed by combining multiple instances that were actually the same using a Python pipeline.
   b. **Overlap**: A single instance ID was assigned to different instances of the same instrument class that overlapped in pixel space. We reannotated these frames using our annotation tool.
   c. **Dataset Noise**: Stray erroneous pixels, incorrect instrument class labels, and missing instrument annotations led to incorrect instance ID assignments. We built pipelines to remove instances caused by dataset noise and correct class labels. Missing instruments were also annotated as required.

Semi-Automatic Annotation.　　For the CholecT50-full dataset, which requires annotations for full sequences (nearly 30,000 frames from 15 sequences extracted at 1 fps), we opted for a semi-automatic approach with a human-in-the-loop, utilizing a deep learning model to assist in producing instance segmentation annotations.

1. Initial Annotation: We began with initial annotations from the Instance-CholecSeg8k dataset. Since CholecSeg8k only contains two tool categories (Grasper and Hook), which is fewer than the seven tool categories identified for annotation, we performed additional annotations using the annotation tool. We focused on images that included the new instruments. The combined dataset of Instance-CholecSeg8k and these initial annotations provided a foundation to start the semi-automatic pipeline for generating labels.
2. Model Training: A deep neural network, the Real-Time Models for Object Detection and Instance Segmentation Large (RTMDet-Ins-l)[20], was trained using the available annotated datasets (Instance-CholecSeg8k and Instance-CholecT50-sparse).
3. Annotation Proposal Generation: Leveraging the trained model, we generated annotations for a larger portion of the unannotated data.
4. Annotation review and correction: Human annotators reviewed the model-generated annotations and corrected any errors or inaccuracies, ensuring the quality and accuracy of the annotations. Minor errors were corrected with the capabilities of the LabelMe labeling tool, while major errors requiring new annotations utilized the point-prompt method of the SAM model.
5. Augment training dataset: The machine learning model was retrained using the expanded training dataset, incorporating the corrected annotations. This iterative process enhanced the model's performance. The model was trained twice: initially, with an annotation set of approximately 11k images and their corresponding labels. The trained models was used to generate labels for about 10k images. Human annotators corrected labels for 5k of these images. The model was then retrained with 21k images, generating labels for another 17k images, with labels for 4k images requiring human annotators for correction.

A visual representation of our semi-automatic approach to annotation can be seen in Fig. 3.
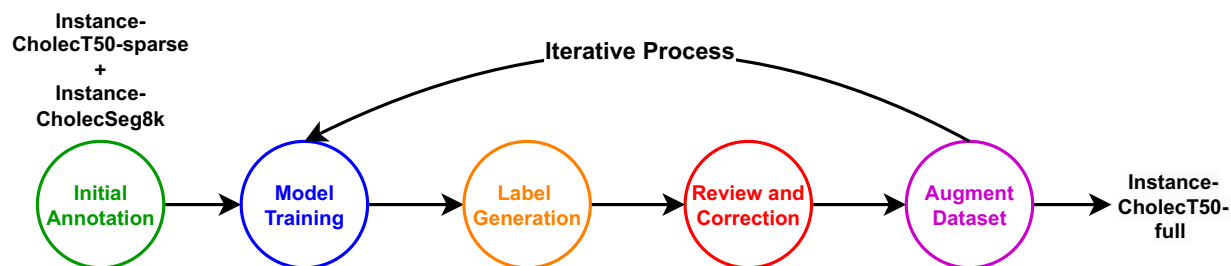
Fig. 3 Semi-automatic annotation pipeline. The process begins with initial annotations using Instance-CholecT50-sparse and Instance-CholecSeg8k, followed by model training. The trained model generates labels, which are then reviewed and corrected by human annotators. The corrected annotations augment the training dataset, iterating through the cycle until the final Instance-CholecT50-full dataset is produced.

Annotation Workflow and Procedures. This section provides an overview of the step-by-step procedure used to create the CholecInstanceSeg dataset, along with detailed insights into the human effort involved in the annotation process. We organize the discussion into two main topics: the step-by-step procedure, and Annotators.

Step-by-Step Procedure.

1. Annotation of Instance-CholecSeg8k (2 weeks): Converted the CholecSeg8k dataset, which initially contained semantic segmentation labels, into Instance-CholecSeg8k with instance IDs. Quality checks were performed to ensure accuracy. This dataset partition only had two classes and lacked many challenging scenarios present in other datasets.
2. Creating the Labelling Protocol (3 weeks): Extensive discussions among the annotation team addressed the challenges encountered during the initial annotation attempts of the Instance-CholecT50-sparse partition. These discussions led to the development of a comprehensive labelling protocol to handle hard cases and unique scenarios effectively and consistently.
3. Annotation of Instance-CholecT50-Sparse (3 weeks): Annotated the Instance-CholecT50 dataset partition using the annotation tool. This occurred in tandem with the creation of the labelling protocol. Quality checks ensured annotations were accurate and consistent with the newly established guidelines.
4. Annotation of Instance-CholecT50-Full (3 weeks): Using Instance-CholecT50-Sparse and Instance-CholecSeg8k as initial annotations, we employed a semi-automatic pipeline to generate the Instance-CholecT50-Full dataset. This process included iterative model training and corrections to enhance annotation quality.
5. Annotation of Instance-Cholec80 (1 week): To ensure comprehensive coverage and enhanced diversity for CholecInstanceSeg, we utilized the annotation tool to manually annotate the Instance-Cholec80 partition.
6. Final Quality Control (2 weeks): Quality control was conducted after the creation of each dataset and again upon finalizing CholecInstanceSeg. This process included several steps to ensure accuracy and consistency. Initially, we performed manual quality control, manually reviewing each image to ensure correct annotations. A second annotator then reviewed a subset of CholecInstanceSeg and performed additional corrections.

We also leveraged specific properties of CholecInstanceSeg and its relationships with other datasets to enhance quality control. For instance, we used existing tool presence labels from related datasets for cross-validation. Detailed steps for quality control, utilizing these properties and external dataset labels, are extensively discussed in the quality control section.

Annotators. The annotation process for creating the CholecInstanceSeg dataset involved a primary annotator responsible for the majority of the annotation tasks and quality control. A secondary annotator, with greater experience and medical qualifications, assisted with annotation and quality control. Both annotators were supported by an expert team to ensure accuracy and consistency in the annotations.

Data Records

The CholecInstanceSeg dataset[21] is publicly available. Cholec80 and CholecT50 video frames used to generate the dataset can be obtained from the CAMMA research group website(http://camma.u-strasbg.fr/datasets). CholecSeg8k can be obtained at (https://www.kaggle.com/datasets/newslab/cholecseg8k).

The dataset is provided as a single archive, *CholecInstanceSeg. zip*, which contains the instance segmentation annotation files in *cholecinstanceseg. zip* and a metadata file *cholecinstanceseg_metadata. csv* that provides metadata information about each annotation file and can be used to link each file to its source dataset. The annotation files are structured into two main directories: train and val, each containing individual video sequence folders. Each sequence folder includes an *ann_dir* subdirectory that stores instance segmentation annotations in JSON format. A schematic representation of the dataset folder structure and a sample JSON annotation file are shown in Fig. 4

Annotations are formatted following the LabelMe convention. Each shape is listed under the *shapes* key, with a *label* field specifying the instrument class, *points* containing the (x, y) coordinates of the polygon defining the
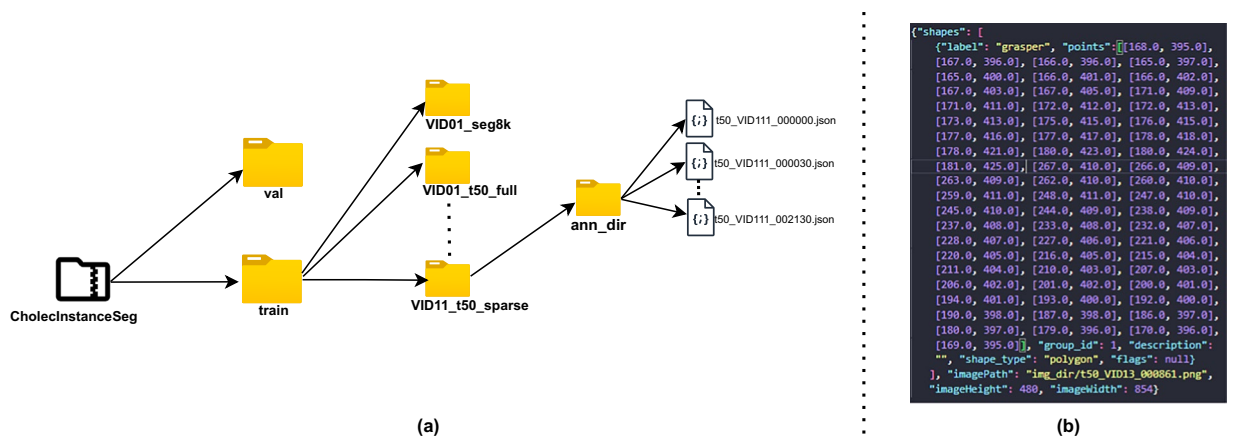
Fig. 4 (**a**) Dataset directory structure for CholecInstanceSeg. Each split (e.g. train) contains subdirectories for each sequence, with further subdirectories for annotations (*ann_dir*). (**b**) Sample JSON annotation file showing the structure of the annotation data, including class labels ("*label*"), polygon information ("*points*") and instance IDs ("*group_id*").
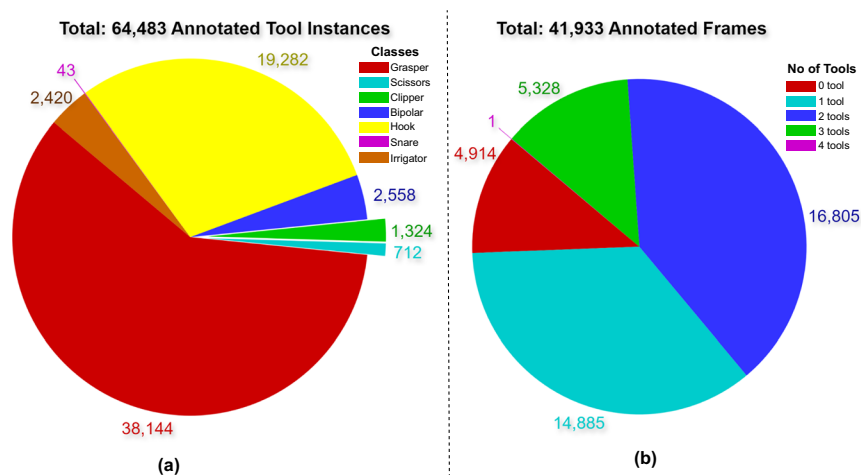

Fig. 5 (**a**) Distribution of each tool class across the dataset. (**b**) Distribution of the number of tools per frame.

object's boundary, *group_id* indicating the instance ID, *image_Height* for the height of image annotated and *image_Width* for its width.

Technical Validation

We provide an in-depth technical validation of the CholecInstanceSeg dataset, demonstrating its reliability and effectiveness for use in research and development of instance segmentation models. We discuss the validation through three main aspects: Data Analysis, training of baseline models, label agreement statistics, and quality control measures.

Data Analysis. *Tool Class Distribution.* Figure 5(a) shows the distribution of tool classes across the entire dataset. Graspers are the most frequently used tools, followed by hooks. Tools like irrigators, clippers, scissors, and bipolars are used less frequently, with snares being the least used. This indicates a high imbalance in tool class occurrences.

Tool Instance Distribution per Frame. 5(b) illustrates the number of tool instances per frame. The majority of frames contain one or two tools, with a maximum of four tools in a single frame. Approximately 5,328 frames (about one-eighth of the dataset) contain no tools.

Partition and Sequence Analysis. Table 2 provides a summary of the number of sequences, number of frames and amount of annotated tools in each of the dataset partitions. Figure 6 provides an analysis of tool instances across different sequences and their partitions. Instance-CholecT50-full has the highest number of tool

	Inst-CholecSeg8k	Inst-CholecT50-sparse	Inst-CholecT50-full	Inst-Cholec80-sparse	Total
No of seq	17(10 shared)	35	15(10 shared)	28	85
No of frames	8,080	2,681	28,317	2,855	41,933
No of tools	10,523	4,098	45,221	4,641	64,483

Table 2. Summary of Dataset Partitions in CholecInstanceSeg.
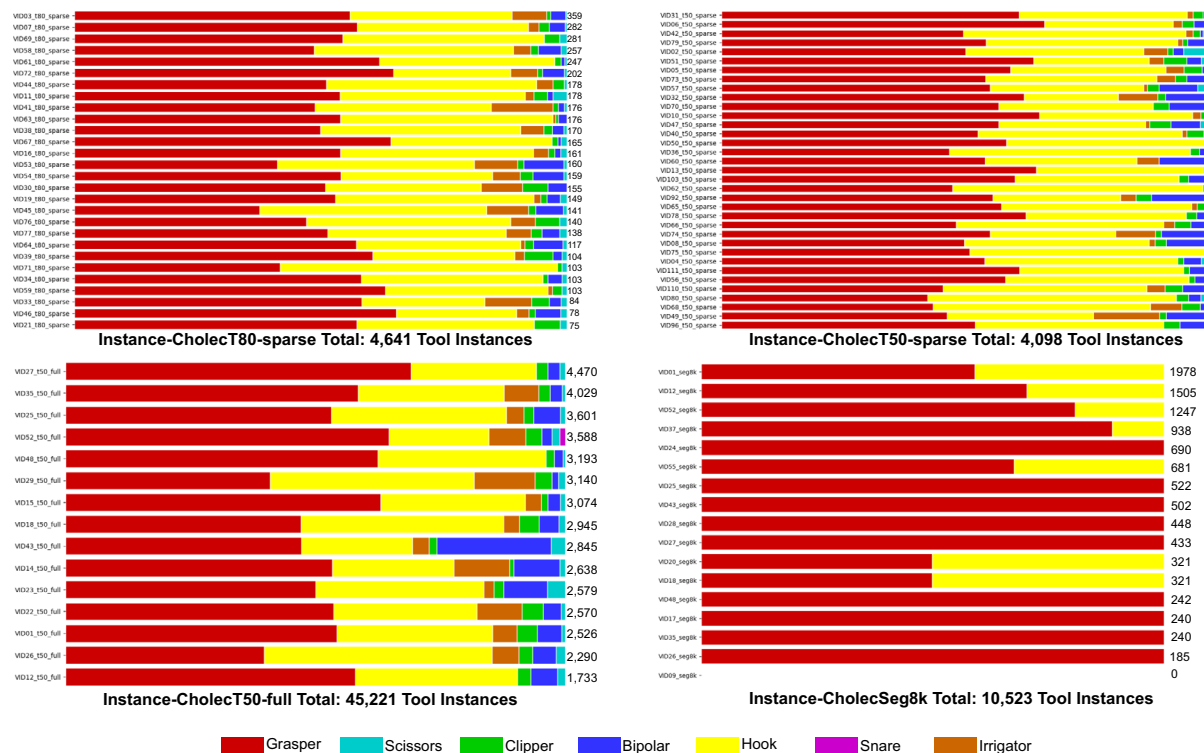
Fig. 6 Distribution of tool instances across sequences and partitions in CholecInstanceSeg.

instances, whereas Instance-CholecT50-sparse has the fewest. This figure highlights the imbalance between densely and sparsely annotated sequences and the presence of a sequence with no tools in Instance-CholecSeg8k.

Training and Evaluation of Baseline Models. *Models.* To validate the usability of the CholecInstanceSeg dataset, two baseline instance segmentation models, The Mask R-CNN[22] and the Mask2Former[23] were trained. Since the objective was to establish benchmark performance metrics that future researchers can use for comparison, we decided to utilize more generalizable baseline models instead of optimized tool-specific models like[24,25]. These models were trained using the training split of CholecInstanceSeg with each model's default hyperparameters.

Recommended Data Splits. To facilitate effective training, validation, and testing of neural networks for instance segmentation, we divide CholecInstanceSeg into standard training, validation, and testing subsets. This division considers the partitions and shared sequences within the dataset.

For the training dataset, we included the entire Instance-CholecT50-sparse partition, which consists of 35 sequences with 2,681 frames. This partition offers significant diversity in surgical scenes due to its varied sequence content. Additionally, 10 in Instance-CholecSeg8k and the 10 sequences in Instance-CholecT50 which come from the same video (shared) were included in the training dataset to avoid redundancy in the validation and testing subsets.

The validation dataset contains 3 out of 7 sequences which are unique to Instance-CholecSeg8k. Similarly, from Instance-CholecT50-full, 1 out of 5 sequences which are unique to Instance-CholecT50 were included in the validation dataset. Additionally, half of the sequences from Instance-Cholec80-sparse were added to the validation dataset to bolster its diversity.

The test dataset includes the remaining 4 sequences from Instance-CholecSeg8k and the remaining 4 sequences from Instance-CholecT50-full. The other half of the sequences from Instance-Cholec80-sparse were also included in the test dataset. This division ensures that the test set includes a wide range of scenarios (sequences), providing a comprehensive assessment of model performance.
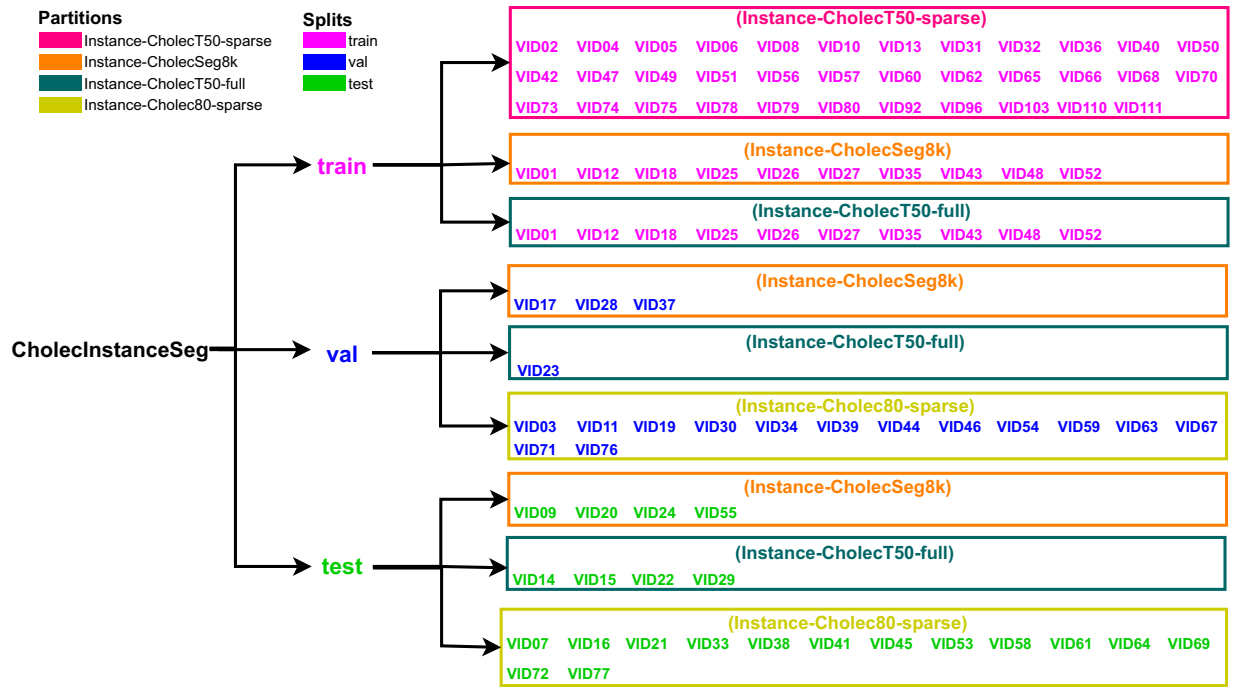
Fig. 7 Recommended Dataset Splits for CholecInstanceSeg. The diagram shows the distribution of sequences across the training, validation, and testing datasets, highlighting the specific partitions (Instance-CholecT50-sparse, Instance-CholecSeg8k, Instance-CholecT50-full, Instance-Cholec80-sparse) and their corresponding sequences.

The final split for the dataset is as follows: the training set contains 55 sequences (26,830 frames), the testing set includes 22 sequences (10,819 frames), and the validation set comprises 18 sequences (4,284 frames).

A visual representation of the recommended dataset split can be seen in Figure 7. We provide scripts to convert the downloaded zip file into a train, test, and validation format based on these recommended splits.

Recommended Metric. To evaluate our models we used the standard COCO mean average precision metric (COCO mAP r@[0.50:0.95]).

From the dataset analysis, we could tell that our dataset has a class imbalance which is captured by the mean average precision, however, it also has an imbalance among sequences, such that some sequences have a lot of tool instances and images while others do not. Hence we wanted to also have a metric that captured how well models are performing on sequences.

COCO smAP involves a slight modification of the COCO mAP metric. The sequence average precision(sAP) for a class j over the whole dataset (sAP_j) in COCO smAP can be written as

$$sAP_j = \frac{\sum_{s=0}^{N} \int_0^1 p_{seq=s}(j, r) \, dr}{N}$$

where s represents the current sequence, N represents the number of sequences evaluated, and $p_{seq=i}(j, r)$ represents the precision at recall r for class j over a sequence s. Essentially sAP_j is calculated by taking the mean of the COCO average precision for that class for each sequence.

To measure variability in performance across sequences, we introduced the standard deviation of sequence average precision (σ_{sAP_j}).

$$\sigma_{sAP_j} = \sqrt{\frac{1}{N} \sum_{s=0}^{N} \left(\left(\int_0^1 p_{seq=s}(j, r) \right) dr - sAP_j \right)^2}$$

where s represents the current sequence, N represents the number of sequences evaluated, and $p_{seq=i}(j, r)$ represents the precision at recall r for class j over a sequence s. σ_{sAP_j} is calculated by taking the standard deviation of the COCO average precision for that class for each sequence.

Finally, the COCO smAP is calculated as a mean of the sequence average precisions across all classes

Method	GR	HO	IR	CL	BI	SC	SN	mAP
Mask-RCNN[22]	37.25	61.84	43.04	58.73	28.42	20.91	—	41.70
Mask2former[23]	61.10	82.16	68.38	81.30	48.02	27.89	—	61.47

Table 3. Performance of baseline instance segmentation models on CholecInstanceSeg for the mean average precision metric (mAP) providing the average precision for each class. GR - Grasper, HO - Hook, IR - irrigator, CL - clipper, BI - bipolar, SC - scissors, SN - snare.

Method	GR	HO	IR	CL	BI	SC	SN	smAP
Mask-RCNN	41.65 ± 12.74	64.06 ± 9.20	52.50 ± 20.09	64.85 ± 16.20	37.62 ± 20.74	40.38 ± 23.54	—	50.18
Mask2former	66.46 ± 8.52	88.90 ± 7.58	74.51 ± 20.35	84.69 ± 16.88	65.20 ± 25.36	69.76 ± 33.20	—	74.92

Table 4. Performance of baseline instance segmentation models on CholecInstanceSeg for the sequence mean average precision (smAP) providing the sequence average precision for each class j (sAP_j). GR - Grasper, HO - Hook, IR - Irrigator, CL - Clipper, BI - Bipolar, SC - Scissors, SN - Snare. The Standard deviation of the sequence average precision σ_{sAP_j} is provided using the ($\pm$) symbol.

$$smAP = \frac{1}{C}\sum_{j=1}^{C} sAP_j$$

We utilize the official COCO implementation present here for calculating the COCO mAP and the metric COCO smAP metric.

Results. The performance results, presented in Table 3 and Table 4, validate the dataset's effectiveness in training robust instance segmentation models.

Notably, the hook class achieved the highest segmentation performance across both models, likely due to its visually distinct features and higher frequency in the dataset. In contrast, the scissor class exhibited the lowest performance, reflecting its smaller representation and potential challenges in detecting this class. Interestingly, the scissor class demonstrated a higher sequence mAP (smAP) but with significant variability (high standard deviation), suggesting that model performance on this class varies considerably across different sequences. Increasing the number and diversity of annotated sequences could improve dataset variability and enhance its applicability to a broader range of surgical scenarios, ensuring fewer seen instruments are better recognized by trained models.

Label Agreement. To ensure the accuracy and consistency of the annotations in CholecInstanceSeg, an analysis of label agreement statistics was conducted. Two types of label agreement were calculated. The inter-annotator agreement and manual vs semi-automatic agreement.

Inter-annotator Agreement. The inter-annotator agreement was assessed using the Panoptic Quality Metric[26]. Two annotators independently annotated a subset of the dataset, and their annotations were compared. The analysis showed a high level of agreement, with a Panoptic Quality score of 91.2, indicating strong consistency between the annotators. The official implementation of the Panoptic Quality metric, available through the panopticapi repository, was utilized for this evaluation.

Manual vs Semi-automatic Agreement. To validate the consistency of our semi-automatic annotation pipeline against the manual annotation process, we measured the agreement between the two methods using the Panoptic Quality Metric. This comparison was crucial to ensure that the increased speed of the semi-automatic pipeline did not compromise annotation quality. The results demonstrated a high level of agreement, with a Panoptic Quality score of 95.7, indicating that the semi-automatic annotations were consistent with the manual annotations.

Quality Control. Our quality control process involved a series of techniques designed to ensure the accuracy and consistency of the CholecInstanceSeg dataset. In addition to manual quality control and a secondary review by another annotator, we leveraged specific properties of CholecInstanceSeg and tool presence labels from Cholec80. The following steps were employed for quality control:

1. checking images with more than 3 tool instances: During annotation, we observed that the most crowded scenes typically included a maximum of three separate tool instances. Therefore, we reviewed all images with more than three tool instances to identify and correct any noise. This process revealed 26 frames with errors due to noise, which were subsequently rectified.
2. utilizing tool presence labels from Cholec80: Tool presence labels from the Cholec80 parent dataset, although noisy and incomplete, were used for cross-referencing. These labels indicate which tools are present in an image but do not specify the number of tools. We cross-referenced our annotations with these tool

presence labels to identify frames where an expected tool class from the tool presence labels was missing in CholecInstanceSeg labels. This method identified 267 frames with errors, achieving an approximately 48% success rate in detecting inaccuracies.

3. reducing noise for scenes with tool overlap: To reduce labelling noise for scenes that include tools occluding other tools, we checked instances that had overlaps. We set a fault tolerance of a 0.1 IoU overlap threshold between tools. All images with intersections exceeding this threshold were reviewed and corrected, resulting in the adjustment of 247 frames.

Code availability

The relevant code for the use of this dataset is found here. This repository includes scripts that provide information about class IDs, sequences, and splits. It also contains notebooks for converting between dataset partitions and splits, utilities for converting to the COCO format[27], visualization tools, retrieving the input images, and additional resources. The official implementation of COCO mAP and panoptic quality metric were used and they can be found on the COCO dataset GitHub page.

References

1. Fuchs, K. Minimally invasive surgery. *Endoscopy* **34**, 154–159 (2002).
2. Islam, M., Atputharuban, D. A., Ramesh, R. & Ren, H. Real-time instrument segmentation in robotic surgery using auxiliary supervised deep adversarial learning. *IEEE Robotics and Automation Letters* **4**, 2188–2195 (2019).
3. Vercauteren, T., Unberath, M., Padoy, N. & Navab, N. CAI4CAI: The rise of contextual artificial intelligence in computer-assisted interventions. *Proceedings of the IEEE* **108**, 198–214 (2020).
4. Ward, T. M. *et al.* Computer vision in surgery. *Surgery* **169**, 1253–1256 (2021).
5. Bodenstedt, S. *et al.* Comparative evaluation of instrument segmentation and tracking methods in minimally invasive surgery (2018).
6. Allan, M. *et al.* 2017 robotic instrument segmentation challenge. Eprint arXiv:1902.06426 (2019).
7. Allan, M. *et al.* 2018 robotic scene segmentation challenge. Eprint arXiv:2001.11190 (2020).
8. Wang, Z. *et al.* 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: 25th International Conference, Singapore, September 18–22, 2022, Proceedings, Part VII*, 486–496 (Springer, 2022).
9. Ross, T. *et al.* Robust medical instrument segmentation challenge 2019. Eprint arXiv:2003.10299 (2020).
10. Psychogyios, D. *et al.* SAR-RARP50: Segmentation of surgical instrumentation and action recognition on robot-assisted radical prostatectomy challenge. *arXiv preprint arXiv:2401.00496* (2023).
11. Hong, W. Y. *et al.* Cholecseg8k: A semantic segmentation dataset for laparoscopic cholecystectomy based on Cholec80. Eprint arXiv:2012.12453 (2020).
12. Twinanda, A. P. *et al.* Endonet: a deep architecture for recognition tasks on laparoscopic videos. *IEEE transactions on medical imaging* **36**, 86–97, https://doi.org/10.1109/TMI.2016.2593957 (2016).
13. Nwoye, C. I. *et al.* Rendezvous: Attention mechanisms for the recognition of surgical action triplets in endoscopic videos. *Medical Image Analysis* **78**, 102433 (2022).
14. Hassler, K. R., Collins, J. T., Philip, K. & Jones, M. W. *Laparoscopic cholecystectomy* (StatPearls Publishing, 2023).
15. Ríos, M. S. *et al.* Cholec80-cvs: An open dataset with an evaluation of strasberg's critical view of safety for ai. *Scientific Data* **10**, 194 (2023).
16. Twinanda, A. P., Yengera, G., Mutter, D., Marescaux, J. & Padoy, N. Rsdnet: Learning to predict remaining surgery duration from laparoscopic videos without manual annotations. *IEEE transactions on medical imaging* **38**, 1069–1078 (2018).
17. Wang, H. *et al.* Supplementary material: Towards open-vocabulary video instance segmentation (2023).
18. Wada, K. *et al.* Labelme: Image polygon annotation with python (2016).
19. Kirillov, A. *et al.* Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 4015–4026 (2023).
20. Lyu, C. *et al.* Rtmdet: An empirical study of designing real-time object detectors. *arXiv preprint arXiv:2212.07784* (2022).
21. Alabi, O. *et al.* Cholecinstanceseg: A tool instance segmentation dataset for laparoscopic surgery, https://doi.org/10.7303/syn60239970 (2024).
22. He, K., Gkioxari, G., Dollár, P. & Girshick, R. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, 2961–2969 (2017).
23. Cheng, B., Misra, I., Schwing, A. G., Kirillov, A. & Girdhar, R. Masked-attention mask transformer for universal image segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 1290–1299 (2022).
24. Baby, B. *et al.* From forks to forceps: A new framework for instance segmentation of surgical instruments. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 6191–6201 (2023).
25. Zhou, Z., Alabi, O., Wei, M., Vercauteren, T. & Shi, M. Text promptable surgical instrument segmentation with vision-language models. *Advances in Neural Information Processing Systems* **36**, 28611–28623 (2023).
26. Kirillov, A., He, K., Girshick, R., Rother, C. & Dollár, P. Panoptic segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 9404–9413 (2019).
27. Lin, T.-Y. *et al.* Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, 740–755 (Springer, 2014).

Author contributions

O.A. wrote the original draft, was the primary annotator, and conducted the experiments and tasks that led to this dataset, K.Z.T. was the secondary annotator for this project, assisted with quality control, and the creation of the annotation guide. Z.Z., C.B., M.S., T.V. were part of the annotation team and mediated ambiguous labels. M.S. and T.V. conceived and supervised this project.

Competing interests

TV is a co-founder and shareholder of Hypervision Surgical Ltd, London, UK. The authors declare that they have no other conflict of interest.

Additional information

Correspondence and requests for materials should be addressed to M.S.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.