npj | digital medicine

Article

Published in partnership with Seoul National University Bundang Hospital

https://doi.org/10.1038/s41746-024-01225-2

Development, deployment and scaling of operating room-ready artificial intelligence for real-time surgical decision support

Sergey Protserov[1,2,3], Jaryd Hunter[1], Haochi Zhang[1], Pouria Mashouri[1], Caterina Masino[4], Michael Brudno[1,2,3] ✉ & Amin Madani[4,5] ✉

Deep learning for computer vision can be leveraged for interpreting surgical scenes and providing surgeons with real-time guidance to avoid complications. However, neither generalizability nor scalability of computer-vision-based surgical guidance systems have been demonstrated, especially to geographic locations that lack hardware and infrastructure necessary for real-time inference. We propose a new equipment-agnostic framework for real-time use in operating suites. Using laparoscopic cholecystectomy and semantic segmentation models for predicting safe/dangerous ("Go"/"No-Go") zones of dissection as an example use case, this study aimed to develop and test the performance of a novel data pipeline linked to a web-platform that enables real-time deployment from any edge device. To test this infrastructure and demonstrate its scalability and generalizability, lightweight U-Net and SegFormer models were trained on annotated frames from a large and diverse multicenter dataset from 136 institutions, and then tested on a separate prospectively collected dataset. A web-platform was created to enable real-time inference on any surgical video stream, and performance was tested on and optimized for a range of network speeds. The U-Net and SegFormer models respectively achieved mean Dice scores of 57% and 60%, precision 45% and 53%, and recall 82% and 75% for predicting the Go zone, and mean Dice scores of 76% and 76%, precision 68% and 68%, and recall 92% and 92% for predicting the No-Go zone. After optimization of the client-server interaction over the network, we deliver a prediction stream of at least 60 fps and with a maximum round-trip delay of 70 ms for speeds above 8 Mbps. Clinical deployment of machine learning models for surgical guidance is feasible and cost-effective using a generalizable, scalable and equipment-agnostic framework that lacks dependency on hardware with high computing performance or ultra-fast internet connection speed.

Complications from surgery are one of the most significant sources of morbidity, mortality and costs in health care, with more than 3 million patients suffering a major preventable operative complication every year[1–3]. This problem is especially pronounced in remote, rural and developing regions of the world, where almost 5 billion people lack access to surgical expertise and basic surgical care, resulting in significant inequities[4]. Most adverse events during surgery tend to be caused by errors in cognitive processes by the surgical team, such as a lapse in visual perception or situation awareness that subsequently lead to decisions, actions or behaviors that lead to such events (e.g. surgeon inadvertently dissecting in the wrong anatomical plane and injuring an important structure)[5,6]. Therefore one potential method to improve the quality and safety of surgery is through

[1]DATA Team, University Health Network, Toronto, ON, Canada. [2]Department of Computer Science, University of Toronto, Toronto, ON, Canada. [3]Vector Institute for Artificial Intelligence, Toronto, ON, Canada. [4]Surgical Artificial Intelligence Research Academy, University Health Network, Toronto, ON, Canada. [5]Department of Surgery, University of Toronto, Toronto, ON, Canada. ✉e-mail: brudno@cs.toronto.edu; amin.madani@uhn.ca

methods that provide surgical teams with real-time data-driven assistance to avoid errors.

Deep learning artificial intelligence (AI) algorithms have demonstrated the ability to achieve expert-level perception and identification of anatomy in the surgical field[7–11]. These computer vision tasks can range from image-level classification to pixel-level semantic and instance segmentation. To date, several models have been developed, mostly for minimally-invasive surgeries with various levels of validation and testing. In previous work a PSPNet-based segmentation model with a ResNet-50 backbone was trained to predict safe and dangerous zones of dissection in laparoscopic cholecystectomy surgery videos[12–14]. Owen et al. were able to enhance the model training process through the integration of label-relaxation and self-supervised steps[15,16]. Jaafari et al. focused on classification, where their team trained a model to identify any of seven surgical tools that might be present within a surgical video frame, and boosted their model's performance via ensemble training[17]. Another area of interest follows the work done by Twinada et al., where their team focused on using AI for surgical phase recognition in a surgery video[18]. While these models have shown tremendous promise of a new era of digital surgery to boost surgical judgment, they remain proofs-of-concept that have achieved acceptable quality metric values on testing data, and the corresponding studies have not addressed many of the technical challenges that need to be solved for clinical deployment.

Specifically, there are two major aspects that might limit dissemination. Firstly, AI models need to be trained and configured in a way that would allow for deployment in a variety of operating theaters with different surgical tools, optimizing for performance and generalizability[19]. Pre-processing techniques need to account for real-world datasets from many institutions that are acquired from different surgical platforms and manufacturers, each using specific camera configurations and video processing units that result in a high variety of video resolution, aspect ratio and frame rate. Typically, images used as model inputs are resized to a uniform shape, which, in the case of heterogeneous aspect ratios, can lead to distortion of geometry, data skew and bias[10,15,18]. Combining images obtained from different cameras often leads to such shape heterogeneity.

A second consideration for the deployment of surgical AI is adapting the model and the hardware to the computational resources that are typically available in most hospitals. The majority of existing approaches use deep residual backbones such as ResNet, AlexNet and their modifications[10,15,18,20]. While this results in higher accuracy, one major limitation is that they run much slower than lighter architectures, making them impractical for real-time segmentation during live surgeries, while deployment through a cloud environment would be further subject to bandwidth and latency constraints on the hospital's internet connection. To our knowledge, no algorithm for surgical videos has shown to scale for intraoperative decision support without requiring Graphics Processing Units or other customized, and often expensive, hardware at the hospital where the surgery is taking place[7–11]. Most remote and developing regions also often lack robust internet connectivity and have limited access to the necessary infrastructure, thereby further exacerbating existing inequities. These are critical obstacles that have yet to be addressed by any proposed solution should AI ever be expected to democratize surgical care by disseminating expertise to resource-limited settings.

In this study we use laparoscopic cholecystectomy, one of the most commonly performed operations[21,22], as a use case for developing and testing machine learning models and subsequently developing and testing a data pipeline that will enable real-time use from any geographic location on pervasive devices. Major bile duct injuries are devastating examples of adverse events that can occur during cholecystectomy and these are often due to errors in judgement that result in surgeons dissecting in anatomical planes that have a high risk of causing such injuries[5]. To augment decision-making, the GoNoGoNet algorithm was previously developed to perform semantic segmentation tasks on frames of surgical videos in order to highlight regions of dissection where there is a low risk of injury and it is safe to dissect ("Go zones") and regions where there is a high risk of causing injury ("No-Go zones")[12]. Since its initial development and preliminary testing, this model and its subsequent modifications have undergone external validation amongst panels of experts[23] and have also been shown to make predictions that could potentially avoid bile duct injuries[24].

Using Go and No-Go zone semantic segmentation during laparoscopic cholecystectomy as a use case, this study develops and tests the performance of a novel data pipeline linked to a web platform that will enable model use on live surgical video streams from any device and a wide range of internet connection speeds, including geographic locations with low connectivity.

Results

Study design

In the first phase of this study, several lightweight deep neural network models that are amenable for scaling and real-time inference were trained to predict Go and No-Go zones on laparoscopic cholecystectomy images, and then validated and tested. In the second phase of the study, a machine learning data pipeline was created to enable these models to provide real-time inference on any stream of surgical video via a web platform, accessible from any pervasive edge device such as laptop computers, desktop computers or mobile devices.

Model architectures

The machine learning task was formulated as a pixel-level semantic segmentation of Go and No-Go zones. Instead of the original GoNoGoNet model, which used a PSPNet architecture with a ResNet-50 backbone[12,23], two new models were trained that are more suitable for real-time inference. The first was a well-established U-Net architecture with some modifications, such as incorporating dropout, batch-normalization layers and using a non-standard PReLU activation function.[14,25–28]. The U-Net architecture implementation was adapted from Buda's published work[29]. While many semantic segmentation models in medical imaging have previously used other more complex architectures, typically by making use of heavy backbones, we instead used an architecturally-simple model with no backbone to optimize inference speed for real-time segmentation, and to make it easier for the model to train from limited amounts of data without overfitting.

To counteract large variabilities in frame shapes and sizes, we downsized all input images to a fixed height (128) while preserving the aspect ratio. For the U-Net based model, instead of training on a full frame, each frame is broken down into overlapping patches and training is done on each patch independently. In the inference pipeline, an average of the predictions on the overlapping regions is taken when reassembling the full frame prediction from the patch-level predictions. Such a pipeline removes the reliance on a fixed frame size or aspect ratio as frames can be split into any number of fixed-size patches, and thus is compatible with any laparoscopic instrument (camera) or operating theatre equipment. This patch-level architecture also limits the amount of frame-level biases that could exist in the data by only presenting parts of the input to the model at a time.

The second model used a SegFormer architecture, using its implementation from HuggingFace Transformers library, with random weight initialization[30,31]. With this model, we accounted for different input shapes by downsizing inputs to a height of 128 and then padding them to a fixed shape, regardless of the original aspect ratio. Figure 1 depicts a schematic of both the model architectural and network pipelines, and Supplementary Note 1 summarizes the pre-processing pipeline.

Two different datasets were used in this study (Dataset 1 and Dataset 2; see Methods). Dataset 1 was split into training, validation and testing subsets in proportions 70%/15%/15% respectively. The splits were performed on a per-case basis for every institution independently to avoid data leakages across subsets and ensure that different institutions are equally well-represented in training, validation and testing subsets. We subsequently re-tested the model on Dataset 2, which is an independent multicenter dataset with each frame annotated by an external panel of expert surgeons, who did not annotate Dataset 1. This assessment served to demonstrate the models' generalization capabilities and any potential performance degradation when run against videos from previously unseen institutions. Performance results are reported for both the Dataset 1 testing set and the entire Dataset 2.

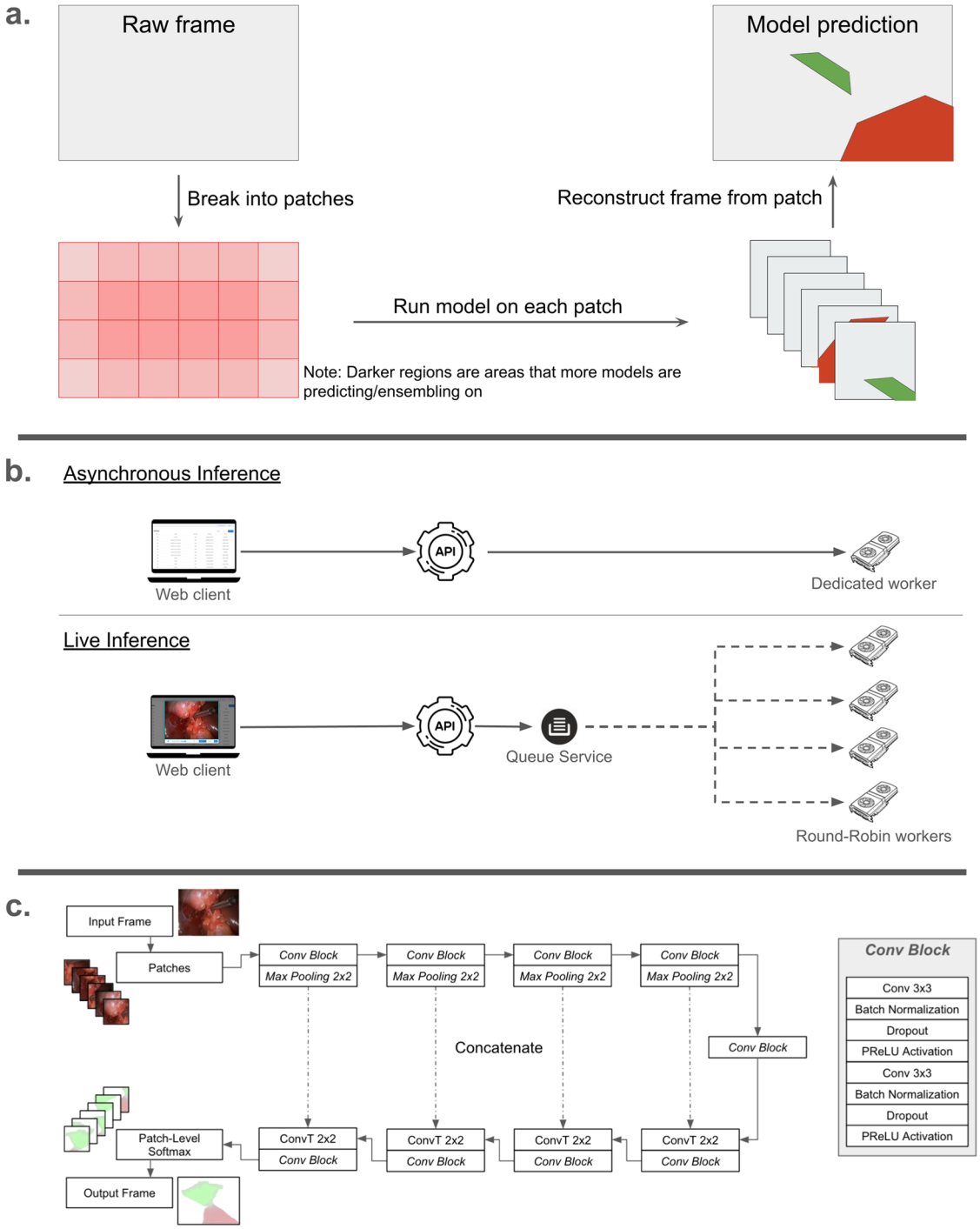

Fig. 1 | Details of model architecture, pipeline, and network framework. a shows the network framework for both live (synchronous) and asynchronous modes of inference. **b** shows how training and inference on patches are performed for the patch-based U-Net model. **c** shows the U-Net patchwork architecture.

Model evaluation

Both U-Net and SegFormer models underwent hyperparameter tuning, both for model and training parameters. The best model of each architecture was selected by cross-entropy loss on the validation subset, where each class was weighed inversely in relation to its frequencies in the training subset. The best-performing U-Net model had bias terms in convolutional layers, extracted 16 features in the first convolutional layer and doubled this amount at every subsequent encoder layer, used a PReLU activation function and a dropout probability of 0.2. The best-performing SegFormer model was found to be the MiT-b0 hyperparameter configuration preset, as implemented in HuggingFace Transformers. Both models were tested and deployed onto the web-platform.

The U-Net model achieved mean Dice score of 57% (+/− 0.05) and 76% (+/− 0.04), precision of 45% (+/− 0.04) and 68% (+/− 0.05), recall of 82% (+/−0.07) and 92% (+/− 0.04), and RAE of +92% (+/− 0.17) and +47% (+/− 0.17) for predicting Go and No-Go zones on Dataset 2, respectively. The model incorrectly predicted a Go zone pixel as a No-Go zone at a rate of 12%, while the more dangerous mistake of predicting a No-Go zone as a Go zone is limited to only 4%. The SegFormer model performed similarly on Dataset 2, achieving a Go and No-Go zone

Table 1 | Segmentation performance of the models, reported at multiple thresholds for Go zone prediction

Dataset	Threshold	Region	Dice	RAE	Precision	Recall
Dataset 1	33% (Default)	No-Go	0.80 ± 0.02 0.75 ± 0.02	0.29 ± 0.07 0.21 ± 0.08	0.76 ± 0.02 0.81 ± 0.02	0.88 ± 0.02 0.88 ± 0.02
		Go	0.67 ± 0.01 0.71 ± 0.02	0.78 ± 0.07 0.38 ± 0.05	0.55 ± 0.02 0.64 ± 0.02	0.91 ± 0.01 0.84 ± 0.02
	50%	Go	0.67 ± 0.01 0.71 ± 0.02	0.67 ± 0.06 0.38 ± 0.05	0.57 ± 0.02 0.64 ± 0.02	0.88 ± 0.02 0.84 ± 0.02
	66%	Go	0.68 ± 0.02 0.71 ± 0.02	0.22 ± 0.05 0.23 ± 0.04	0.67 ± 0.02 0.68 ± 0.02	0.77 ± 0.02 0.79 ± 0.02
	80%	Go	0.62 ± 0.02 0.70 ± 0.02	−0.25 ± 0.05 0.07 ± 0.04	0.78 ± 0.02 0.72 ± 0.02	0.57 ± 0.03 0.73 ± 0.02
	90%	Go	0.42 ± 0.03 0.67 ± 0.02	−0.65 ± 0.03 −0.12 ± 0.04	0.88 ± 0.02 0.76 ± 0.02	0.31 ± 0.02 0.65 ± 0.02
Dataset 2	33% (Default)	No-Go	0.76 ± 0.04 0.76 ± 0.05	0.47 ± 0.17 0.46 ± 0.16	0.68 ± 0.05 0.68 ± 0.05	0.92 ± 0.04 0.92 ± 0.04
		Go	0.57 ± 0.05 0.60 ± 0.05	0.92 ± 0.17 0.48 ± 0.15	0.45 ± 0.04 0.53 ± 0.05	0.82 ± 0.07 0.75 ± 0.07
	50%	Go	0.53 ± 0.06 0.60 ± 0.05	0.67 ± 0.17 0.47 ± 0.15	0.43 ± 0.05 0.54 ± 0.05	0.72 ± 0.09 0.75 ± 0.07
	66%	Go	0.50 ± 0.08 0.58 ± 0.06	0.13 ± 0.15 0.28 ± 0.16	0.47 ± 0.07 0.57 ± 0.06	0.58 ± 0.10 0.69 ± 0.08
	80%	Go	0.43 ± 0.09 0.56 ± 0.07	−0.39 ± 0.11 0.10 ± 0.15	0.55 ± 0.10 0.58 ± 0.07	0.38 ± 0.09 0.63 ± 0.09
	90%	Go	0.24 ± 0.08 0.53 ± 0.08	−0.78 ± 0.07 −0.11 ± 0.15	0.60 ± 0.12 0.64 ± 0.06	0.17 ± 0.06 0.55 ± 0.09

The U-Net results are presented as the top row of each cell, and the SegFormer results are presented as the bottom row.

(respectively) mean Dice score of 60% (+/− 0.05) and 76% (+/− 0.05), precision of 53% (+/− 0.05) and 68% (+/− 0.05), recall of 75% (+/−0.07) and 92% (+/− 0.04), and RAE of +48% (+/−0.15) and +46% (+/− 0.16), with the more dangerous mistake of predicting a No-Go zone as a Go zone being limited to only 1%. Table 1 summarizes model performance for both datasets. Figure 2 shows the row-normalized confusion matrices demonstrating the pixel-level performance of both models.

Web application and infrastructure

It is integral that any platform introduced into a clinical setting incorporates end-users into the workflow and decision-making process. Evidence suggests that one strong limitation for clinicians to use AI for patient care is its lack of explainability and understanding of any underlying uncertainty for a given prediction[32,33]. Establishing this data pipeline with a human-in-the-loop configuration will help overcome these implementation obstacles. We developed a web platform to display the Go/No-Go predictions to surgeons in real-time. The platform is summarized here and described in more details in Methods.

The web platform was designed for an overlay display to show Go zones as shaded green and No-Go zones as shaded red to facilitate usability. Furthermore, the confidence level of a prediction for each pixel is reflected by its transparency such that both Go and No-Go zones are ultimately displayed as heat maps. Finally, a slider was introduced for users to select a threshold for displaying confidence level of Go zone prediction. This is extremely important for real-time guidance during surgery, as surgeons will have significant variation in their risk tolerance for a given procedure, and more importantly, this level of tolerance will change for different patient-specific scenarios. For instance, an emergency laparoscopic cholecystectomy with significant inflammatory changes and challenging anatomy has a higher risk of adverse events. Hence surgeons would potentially want to err on the more conservative side, and select a higher threshold for the model's predicted probability of Go zone. The design presented here provides a flexible system to account for these variations and to allow surgeons to choose a display option that is appropriate based on their risk tolerance for a specific procedure. Ongoing work will further assess end-user perspectives for the entire operating team and inform future design for additional customization.

Data and network pipeline

To make our model accessible on-demand from any geographic location, a web-based platform was developed that allows for inference on streams of surgical videos using pervasive edge devices (e.g. laptop, smartphone) and ensuring compatibility across devices. This platform is described in the Methods section. The platform was tested across varying network speeds for both U-Net and SegFormer models. Under optimal conditions (network bandwidth speeds above 32.0 Mbps) and utilizing a flow-control algorithm to optimize performance[34], the U-Net model was able to handle transmitting a stream at 65.9 (± 0.11) frames per second (fps) with a round-trip delay of 77 (± 0.2) ms. This included model inference (including all pre- and post-processing steps), which took 20.75 ms. The SegFormer model performed similarly, achieving a frame-rate of 65.9 (± 0.10) fps with a total delay of 78 (± 0.3) ms, of which 25.63 ms was the model inference time. However, under slower internet speeds (below 2 Mbps), the platform was only able to deliver a stream at a maximum of 13 fps with a round-trip delay of at least 380 ms (speed test results are presented in Table 2). This may hinder the platform's usability in locations with poor connectivity.

To make our platform applicable to resource-poor regions, we implemented an option that reduces network requirements by downsizing frames on the client-side before sending them to the server, to be upscaled to model-native size on the backend. Similarly, the model's prediction will be downsized before being sent to the client, to be upscaled on the client side. By downsizing frames and predictions by a factor of 2 along each dimension, we transmit 4x less data, allowing for even the slowest internet connection speeds to handle real-time inference. This strategy allows the platform to maintain a prediction stream of at least 60 fps, with a delay under 150 ms, for network speeds as low as 2 Mbps. For the speed of 1 Mbps the platform was able to achieve a frame-rate of 46 fps with an overall delay under 200 ms. Speed test results with this downscaling option are also presented in Table 2. We tested the effect this strategy has on segmentation quality for both models, and the drop of quality (Supplementary Table 2) was less than 2% for Dice, recall and precision for both models. With the downscaling option active, but without flow-control algorithm, at the fastest tested speed (32.0 Mbps) the platform was still able to obtain a frame-rate of 59.0 (± 0.87) fps, however with a delay of 423 (± 0.61) ms, 7-fold slower than

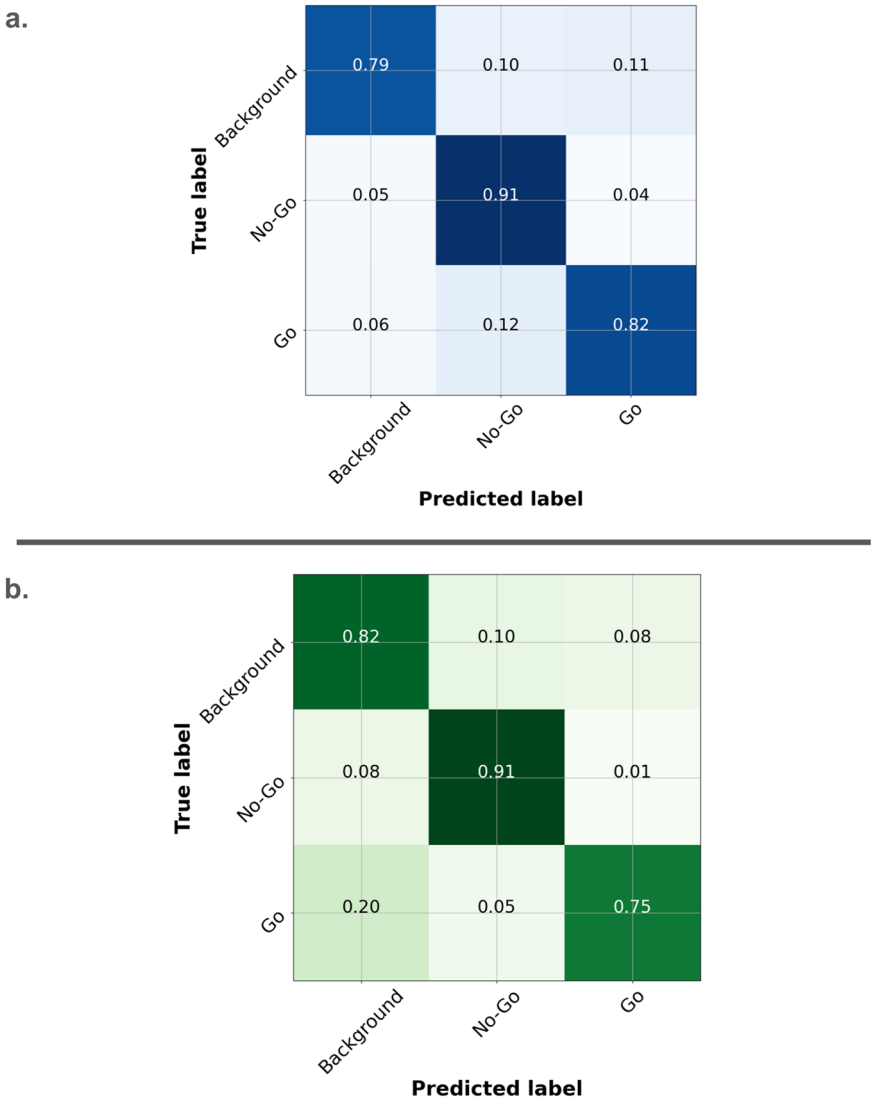

Fig. 2 | Mean row-normalized confusion matrices on Dataset 2. Both U-Net (**a**) and SegFormer (**b**) models are shown using the default 33% confidence threshold for Go zone prediction.

without the flow-control algorithm. Furthermore, delays of over 1500 ms occurred under slower speeds (<8 Mbps). This highlights that flow-control algorithm is instrumental for providing a real-time prediction stream.

Discussion

Advances in data science have permeated various facets of clinical practice and to date, multiple AI algorithms have begun to change how clinicians practice and deliver care to their patients[35]. The applications of machine learning techniques in computer vision have been particularly palpable for surgeons given how many high-stakes decisions take place in the operating room, and how even the slightest deviations in one's mental model during surgery can lead to adverse events with significant downstream impact on patients. The development and use of new models to provide decision support to surgeons for guidance and navigation can be potentially transformative for the field of surgery. Despite the early success of AI algorithms for making predictions on surgical scenes, there are still significant obstacles that need to be overcome before these algorithms are implemented at the point-of-care in the operating room. Meeting mathematical benchmarks of computer vision metrics, while a necessary step for model validation, is insufficient, because most algorithms have not demonstrated the ability to be successfully deployed, potentially due to large and slow model architectures, lack of generalizability to real-world input data, and the necessary equipment needed to run models in real time. Specifically, there is a need to develop sustainable, cost-efficient and scalable solutions to disseminate these models to many operating rooms in various geographic locations, and to ensure they reach those with the greatest need - especially in remote and resource-poor settings that lack infrastructure. Using laparoscopic cholecystectomy as a use case, this study describes a novel framework and methodology to enable intraoperative AI model deployment in a scalable and accessible manner, onto any edge device regardless of geographic location or network connectivity and is agnostic to the use case and surgical procedure.

Surgical video data, originating from platforms that enable image-guided surgery (e.g. laparoscopy) are plagued with significant heterogeneity. These include background noise, variations in patient-specific anatomy and pathology, different instruments used in different settings, different surgical approaches, and other factors such as smoke and fluids being present in the field of view. There is also significant technical variability depending on the platform that is used, such as image quality, resolution, white balance, aspect ratio, zoom, lighting, frame rate, camera angle, and additional data on the monitor. This presents a unique challenge whereby models that are intended for widespread use need to account for these variations.

Current state-of-the-art is not designed for scalability and generalizability, and typically involves architectures that require input data to be resized to a uniform format. Furthermore, these models typically use heavy architectures that have high overhead computational costs, making them impractical for real-time deployment. Significant delay for computer vision tasks leads to pixel segmentation overlays that are not in synchrony with the

Table 2 | Web platform FPS and delay test results at varying network speeds, utilizing flow-control algorithm, with and without downscaling frames and predictions for network transmission

Network Speed (Mbps)	Model	Size	FPS (frames per second)	Delay (ms)
1	U-Net	Original	6.7 ± 0.00	664 ± 1.2
		Downscaled	48.1 ± 0.21	176 ± 0.6
	SegFormer	Original	6.7 ± 0.00	671 ± 2.2
		Downscaled	46.1 ± 0.74	193 ± 0.8
2	U-Net	Original	13.0 ± 0.02	386 ± 0.4
		Downscaled	64.5 ± 0.12	121 ± 0.6
	SegFormer	Original	13.0 ± 0.03	387 ± 1.0
		Downscaled	63.0 ± 0.55	124 ± 0.8
4	U-Net	Original	25.3 ± 0.04	252 ± 0.3
		Downscaled	66.3 ± 0.10	84 ± 0.5
	SegFormer	Original	24.8 ± 0.09	258 ± 0.1
		Downscaled	65.6 ± 0.13	86 ± 0.4
8	U-Net	Original	48.1 ± 0.04	178 ± 0.6
		Downscaled	66.3 ± 0.06	70 ± 0.3
	SegFormer	Original	47.6 ± 0.09	183 ± 0.3
		Downscaled	65.8 ± 0.08	72 ± 0.4
16	U-Net	Original	65.7 ± 0.04	109 ± 0.6
		Downscaled	66.3 ± 0.07	65 ± 0.4
	SegFormer	Original	65.2 ± 0.14	111 ± 0.1
		Downscaled	65.9 ± 0.08	68 ± 0.3
32	U-Net	Original	65.9 ± 0.11	77 ± 0.2
		Downscaled	66.3 ± 0.09	54 ± 0.4
	SegFormer	Original	65.9 ± 0.10	78 ± 0.3
		Downscaled	65.6 ± 0.08	57 ± 0.4

The model inference pipeline (including all pre- and post-processing steps) had a static run-time of 20.75 ms for the U-Net model, and 25.63 ms for the SegFormer model.

video footage thereby creating a bottleneck limiting real-time usability for a dynamic scene such as the surgical field. The framework proposed here addresses these limitations using two approaches: the application of lightweight machine learning models and a highly optimized network data pipeline that dynamically adjusts based on the client's internet connection speeds.

The architectures utilized in this study have been shown to be viable for training lightweight models that can be used across a variety of data input sizes and formats. The patch-based U-Net model was designed to restrict the contextual information of the frame it is predicting on, thus preventing (or at least limiting) location biases that could occur during training. When evaluated against the standalone testing set (Dataset 2), the models presented rather interesting results. The combination of our evaluation metrics helped show that both the Go and No-Go zones' predictions are generally larger than the ground-truth label (i.e. models are over-predicting), where they generally spill over into the background region. However, the analysis also shows that the U-Net model is over-predicting the Go zone at a much higher proportion compared to the No-Go zone, thus resulting in lower Dice scores for the Go zone. More importantly, both models exhibited a very low likelihood of misidentifying a No-Go zone as a Go zone, which is the most dangerous scenario that could occur (i.e. model actively suggesting to the surgeon to dissect in a dangerous region). While an aggressive prediction is generally seen as beneficial for the No-Go region, it is not always the case for the Go region, where conservative predictions are preferred (i.e. avoid recommending dissections in any potentially dangerous tissue planes). Interestingly, the predictions generated by a patch-based U-Net model for the Go zone are affected by probability thresholding to a much larger extent than predictions generated by the SegFormer model (Table 1). We hypothesize that this is due to probability averaging that occurs on patch overlaps, which discourages extremely high or extremely low predicted probabilities.

To optimize the network data pipeline, we utilized a flow-control algorithm that prioritizes minimizing the overall delay by controlling the rate at which frames are being sent over the network. While under sufficiently high internet speeds this has no significant effect, once internet speed drops below approximately 8 Mbps, the added benefit of the flow-control algorithm becomes more apparent.

Another important optimization that enabled us to achieve high fps and low delay even for poor internet speeds is an option to downsize every frame by a factor of 2 along both dimensions, thus reducing the amount of data to transmit by a factor of 4 compared to when transmitting frames of model-native size. Similarly, the generated prediction is sent from the server to client in a downscaled format, to be upscaled on the client side. We tested the effect this strategy has on segmentation quality for both models, and the drop of quality (Supplementary Table 2) was minimal, although other models in the future may show larger degradation when using this method. By using lightweight models we minimized computational overhead that the models introduce and ensured that they can provide real-time inference on network speeds as slow as 1 Mbps. Ultimately end-users can access any model, from any device, from any geographic location with low connectivity, and still be able to run an AI algorithm on a surgical video stream. For remote and low-resource settings, this framework helps bridge inequity gaps and democratize surgical care.

Although the models presented in this study achieved acceptable performance benchmarks on both test datasets, additional methods can further improve results. Notably, the models were trained on singular frames extracted from videos, which limits the temporal context the models can use for their predictions. This often manifests as occasional "jumps" in predicted Go and No-Go zones during inference on videos. Thus, future work will attempt to account for temporal components in the model architecture, such that it can leverage data across time to create more consistent and accurate predictions. However, such an approach needs to balance model performance improvement with increase in inference time that will likely occur, in order to keep the models viable for real-time prediction. There are also numerous potential sources of epistemic and aleatoric uncertainty and ambiguity. Firstly, while our dataset was very diverse and heterogeneous, more data, especially from edge cases (with gangrenous/chronically scarred gallbladders, major bleeding and complications) could be used to improve model performance. Secondly, traditional metrics of computer vision for surgical videos have limitations and it is likely that the true performance or clinical relevance of the model is not fully represented in traditional mathematical benchmarks[36]. Lastly, surgical expertise is an extremely complex construct and despite our best efforts, we have yet to fully characterize it, let alone train AI algorithms to replicate it[6]. This is demonstrated by the fact that multiple experts annotating Go and No-Go zones will never have perfect concordance in their annotations[37–39].

In addition, a few improvements can be performed on the platform. Currently during live inference, every frame is sent to the backend for processing, unless throttled by flow-control. However, frames captured during surgery (often at 30 fps) have very minimal noticeable difference between each incremental frame. Thus, one potential optimization technique to improve segmentation fps and lower overall delay is to capture changes between sequential frames and for any given frame, update the existing prediction mask for the previous frame, without sending the frame to a backend. This method can be implemented directly in the browser using Optical Flow algorithms[40]. However, implementing this method would require tuning of several parameters, including the delta change that triggers a real-inference request from the backend, and the overall frequency of running real-inferences in an effort to prevent run-away segmentation errors from accumulating. Lastly, all experiments and evaluations presented in this study were done using a static environment with access to computing power located on-site. Although the flow-control algorithm helps manage overall delay under varying internet speeds, it cannot lower the natural delay that exists when connecting and streaming from locations further away geographically (e.g. connecting to a server in Canada from Europe). Thus, another future work direction is the migration of this platform to the cloud,

such that spot instances close to a client's geo-location can be leveraged to provide the lowest possible delay no matter where the client is connecting from. By making our method open source, we are encouraging other investigators to further build upon our methodology.

Finally, there are several considerations that are beyond the scope of this study, including platform, usability, design and implementation into the workflow of the operating room, as well as their ramifications on patient outcome and cost-effectiveness. Preliminary data suggests that most surgeons would prefer an "on-demand" system that would engage as needed as opposed to one that is consistently present that can potentially distract from the surgical procedure and interfere with the flow of surgery[41]. When asked about the optimal method of inference display, most surgeons reported that the multicolor heatmap from the original model would cause significant interference both during a live procedure and for postoperative video analysis. Furthermore, the heatmap only allows display of one target structure and incorporating two structures as heatmaps would cause even greater distraction from the underlying video. The simpler approach of just showing Go and No-Go zones as single colors was a more convenient design. Interestingly, it is difficult to predict how these considerations will impact outcomes and this will be the subject of future research. For example, while computer vision algorithms for polyp detection during colonoscopy have shown promising results with improvement in adenoma-detection rate, there remains a paucity of evidence to suggest improvement in oncologic outcomes[42]. Yet, most clinical trials have shown an increase in total procedure duration time, potentially leading to other unintended consequences (i.e. patient, institutional or societal costs, less access to care, etc.)[43]. Measuring overall time delay is multifactorial, and while infrastructure delay (e.g. cloud computing, network connectivity, displaying model output on local edge devices) may play a role, a greater contributor is more likely to be how end users incorporate AI model inference into their mental model and make subsequent decisions based on the advice they obtain. Ultimately, it will also depend on the number of times an algorithm is utilized throughout a procedure (e.g. one-time assessment as a second opinion, or consistent decision support during an operation).

In this paper we presented a framework and methodology for bringing surgical AI models to end-users via a web platform and demonstrated this using U-Net- and SegFormer-based semantic segmentation models for predicting safe and dangerous zones of dissection on videos of laparoscopic cholecystectomy as an example use case. The web platform is compatible with, and can be accessed from, a wide range of pervasive devices, such as laptops, mobile devices and tablets. By using lightweight model architectures and a highly optimized data and network pipeline, we were able to achieve a high frame-rate prediction stream with low round-trip delay even for low internet speeds, demonstrating the potential of using this platform in remote locations with poor internet connectivity.

Methods

Dataset and annotations

The datasets used to train and test the new algorithm include an initial dataset (Dataset 1) of 289 retrospectively collected open-source videos of laparoscopic cholecystectomy from 37 countries (including all continents), 153 surgeons and 136 different institutions[12]. A second dataset (Dataset 2), used exclusively for testing, comprised of 25 prospectively collected videos from 5 countries (Canada, USA, France, Bolivia and Thailand), 9 surgeons and 7 institutions[23]. Dataset 2 is an independent multicenter dataset with each frame annotated by a panel of expert surgeons (separate from Dataset 1) from the Society of American Gastrointestinal and Endoscopic Surgeons' Safe Cholecystectomy Task Force. Table 3 summarizes the dataset characteristics. Each frame was annotated (freehand) by a panel of expert high-volume surgeons for Go and No-Go zones separately. Given the complexity of surgical anatomy (e.g. varying geometry, lack of clear boundaries, hidden under fatty/fibrous connective tissues), we chose pixel level free-hand segmentation of Go and No-Go zones as the most appropriate problem formulation for this use case. Due to variation in target structure annotation amongst surgeons, visual concordance was used to aggregate labels and select a combination of these annotations into a final mask, which served as the ground truth and was used to evaluate the final model (Fig. 3).

Table 3 | Breakdown of acquired datasets, including demographics and general video metadata [Adapted from [12, 23]]

	Dataset 1[12]	Dataset 2[23]
Total videos	289[a]	25[a]
Number of surgeons	153	9
Number of institutions	136	7
Collection timeline	Retrospective	Prospective
Number of countries	37	5
Acute/chronic cholecystitis	127 (44%)	14 (56%)
Lysis of adhesions	63 (22%)	7 (28%)
Total annotated frames	2627	47
Annotated frames per video (mean)	9	2
Expert annotators per frame	4	6
Resolution (range in pixels)	[576 × 416 – 1984×1100]	[2476 × 1770 – 2964 × 1702]
Width-to-height ratios	[1.01– 2.23]	[1.4– 1.83]
Frame rate	[25–60]	[30–60]
Go zone annotator concordance[b]	0.89 (0.08)	0.89 (0.04)
No-Go zone annotator concordance[b]	0.91 (0.07)	0.90 (0.05)

[a]Selection criteria included: 1) laparoscopic recordings where a cholecystectomy was performed (i.e. subtotal cholecystectomy was excluded), 2) video included the dissection of the hepatocystic triangle from the moment the gallbladder infundibulum was initially grasped for retraction until just prior to clipping and dividing cystic structures, 3) the cholecystectomy was not performed using a top-down approach.
[b]Concordance was calculated using a validated visual concordance test methodology[37–39] as the average per-pixel agreement amongst annotators for each frame. The proportion of pixels that have 100% agreement by annotators is calculated for every frame, and the mean value is calculated for each specific target structure.

Previously, several frames were selected uniformly over the time length of every video and manually reviewed for surgical relevance, with irrelevant frames being discarded (e.g. frames where the camera was outside the operating field within the trocar or outside the abdomen, or where the camera view was completely obstructed, e.g. dirty lens)[12,23]. Using best practices in machine learning, we trained numerous models of both U-Net and SegFormer architectures on the training dataset with different model and training hyperparameters, selected the best models within every architecture based on their performance on a validation dataset, and reported final performance results on a testing dataset. In this paper, training, validation and testing datasets were from Dataset 1 (70%/15%/15% split respectively). We subsequently re-tested the model on Dataset 2. This assessment served to demonstrate the models' generalization capabilities.

Web application

To make our model accessible on-demand from any geographic location, a web-based platform was developed that allows for inference on streams of surgical videos using pervasive edge devices (e.g. laptops) and ensuring compatibility across devices. The platform utilizes ReactJS framework (reactjs.org) for the frontend web application and Flask framework (flask.palletsprojects.com) for the backend server.

The two central functions implemented in the platform are: 1) model inference on a live video stream (synchronous mode), and 2) model inference on a pre-recorded uploaded video or image file (asynchronous mode). For the synchronous mode, users select a source video: built-in camera in the edge device, screen-sharing any application or window for a running video, or direct connection from the laparoscopic tower through a USB video

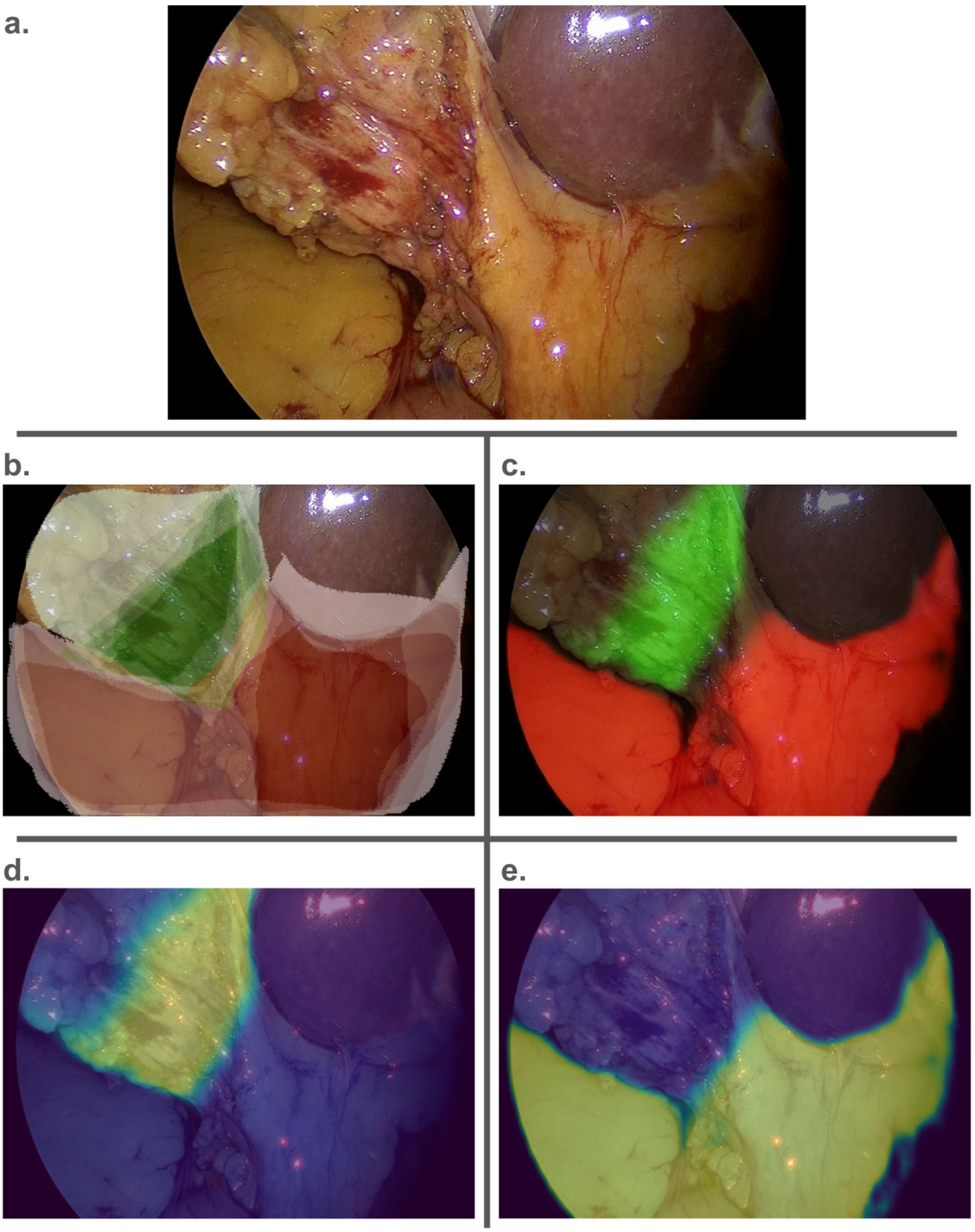

Fig. 3 | Visual representation of an example frame at each stage of the pipeline. This includes **a** the raw frame, **b** ground truth segmentations with overlapping annotations from multiple expert surgeons, **c** Go and No-Go zone model predictions (green and red labels, respectively where brighter color intensity reflect higher model probability), **d** Go zone model prediction as a heat map, and **e** No-Go zone model prediction as a heat map.

capture card. Once selected, the feed is sent to the backend server, which runs model inference on individual frames. Semantic segmentation masks or heatmaps are sent back to the end-user's device and shown as overlay on the raw input video (Fig. 4a). For asynchronous mode, a video file is uploaded to the server, where inference is run, after which it becomes available for the user to download (Fig. 4b).

For each pixel of a given frame, the model predicts the probability for each of the three classes: Go zone, No-Go zone, or neither (background). A pixel will be highlighted as belonging to the class that has the greatest predicted probability. Overlay images are shaded such that the degree of opacity reflects the model's confidence in its prediction. Pixels with a higher probability will have a more opaque green/red colour and pixels with a lower probability will have a more transparent green/red colour.

Successful implementation of surgical guidance systems requires an understanding of surgical judgment, naturalistic decision-making and how surgeons deal with uncertainty in the operating room that may or may not lead to surgical errors. Error tolerance is a highly complex construct of surgical expertise and dependent on numerous patient and pathology-specific factors. To improve end-user utilization and account for uncertainty estimations, a slider was integrated into the platform to allow for dynamic

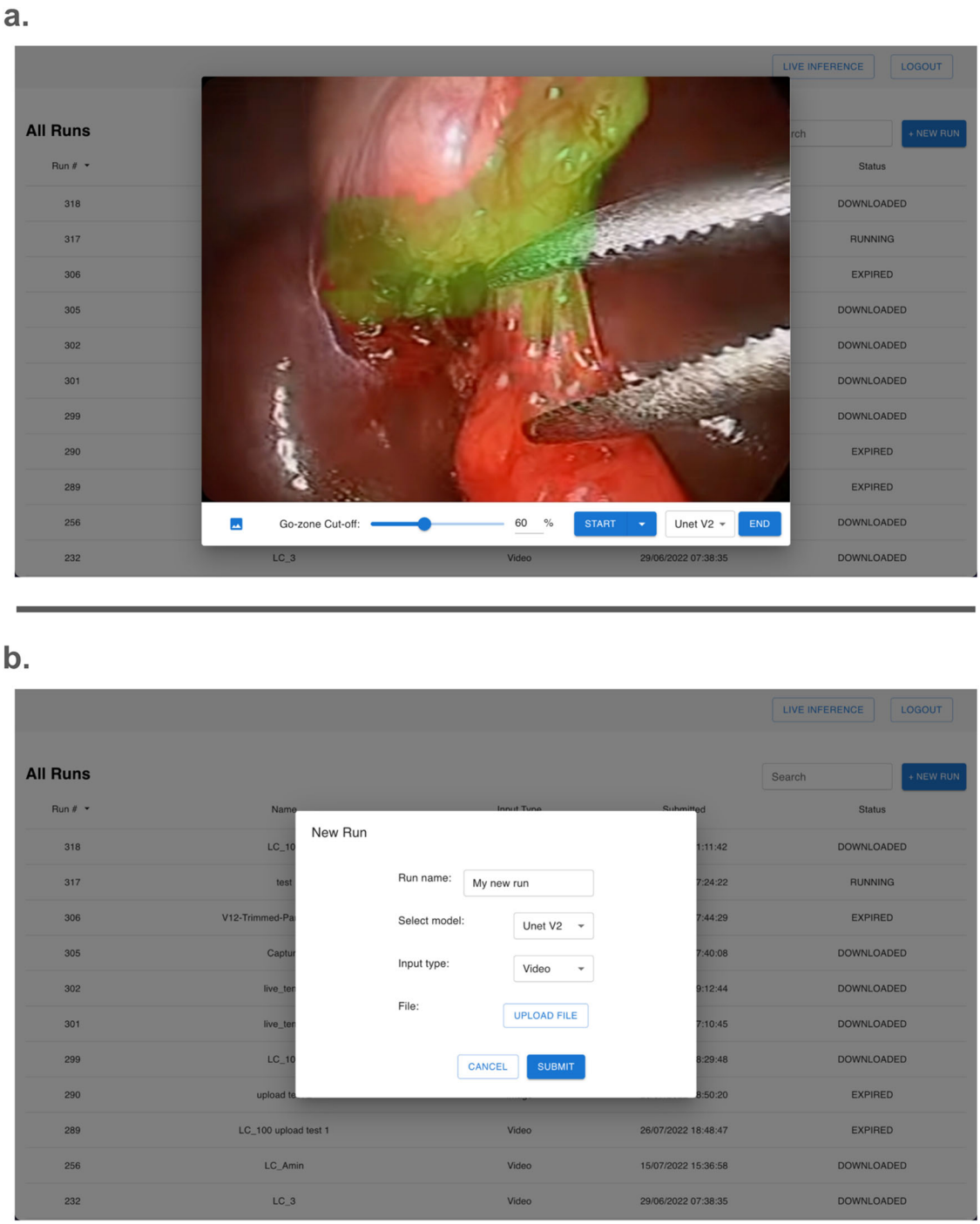

Fig. 4 | Screenshots of the live (synchronous) and asynchronous capabilities available on the platform. For the synchronous mode (**a**), users can select a video stream, specific model and modulate the probability threshold for display. For the asynchronous mode (**b**), users can upload video or image files for prediction, select a model and submit for inference which can be subsequently downloaded after inference is completed.

tuning of the Go-zone's activation threshold, such that pixels for the Go zone are only highlighted if the model's certainty is above a user-selected threshold (Fig. 5). For example, if a 50% threshold is selected, out of pixels for which Go-zone probability is the highest, only those with >50% probability of being in the Go zone will be highlighted as a Go zone pixel. The remainder (Go zone probability is highest, but <50%) will be labelled as the background class.

The platform currently employs an edge-computing approach, where the primary server is located within University Health Network's (UHN, Toronto, Canada) data-center, but is accessible from all intranet & internet connections. The server environment includes a shared pool of four workers on UHN's private cloud, each with a dedicated GPU (NVIDIA Tesla P100). Work is coordinated using a central round-robin queueing system, giving the flexibility to horizontally scale with ease, and allowing for servers to go offline for maintenance without affecting platform usability. The SocketIO protocol was used to allow for bidirectional streaming of data between the client and backend servers and to optimize throughput and delay[44]. Furthermore, each GPU worker was enabled to push directly to the client, reducing the number of network hops and further optimizing throughput. To overcome connectivity issues in resource-poor or remote regions, a flow-control algorithm was also introduced into the synchronous live-inference portion of the platform[34]. This algorithm allows for balancing between the

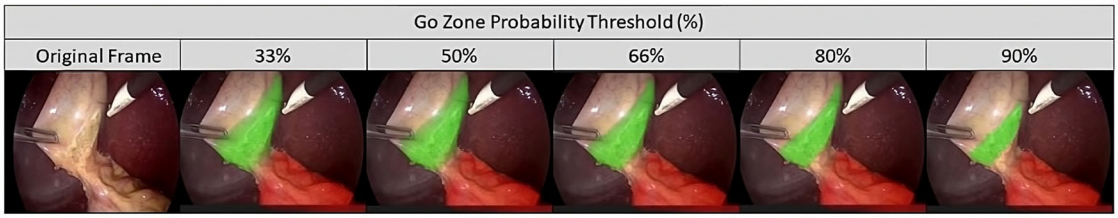

Fig. 5 | Visualization of Go zone display (green) that can be modified according to the probability threshold selected during live inference. Range shows Go zone threshold of 33% (default) up to 90%.

frame-rate and network delay to avoid overwhelming end-users' devices in the event of a poor internet connection. More details on the flow-control algorithm can be found in Supplementary Note 2. Additionally, rendering optimizations were employed in the web-application to optimize performance and minimize any delays on the client device itself. This included modifying the HTML DOM structure to minimize page re-renders when receiving new frame predictions. When measured, the frame rendering component of the platform presented a < 4 ms delay, which was deemed negligible. Lastly, to further minimize network transmission times, an option was introduced to the platform that reduces transmitted data by a factor of 4 by downsizing frames being sent from the client to the server, and the predictions sent back from the server to the client This method was found to provide the best trade-off between network delay and model performance degradation.

Evaluation

Dataset 2 was used for model performance evaluation comparing the model prediction to the ground truth. Four metrics were used: pixel-level precision and recall, Dice Similarity Coefficient (DSC), and Relative-Area-Error (RAE). Details of these metrics are summarized in Supplementary Note 3. Since the raw videos come in a variety of different resolutions, all metrics were calculated for every frame independently and averaged over all frames in the dataset. Results are shown as percentage (+/− standard deviation).

To test the efficacy and delay of the data pipeline and web application, a simulation of various network speeds was performed using the network throttling setting of the Chrome Developer tools. Outcomes included frame-rate (fps) and round trip delay (ms). The tests were performed on a 2020 Macbook Pro laptop connected to the internet via WiFi, using a remote internet connection outside of the UHN intranet network (within Toronto, Canada). Network bandwidth speeds of 1.0, 2.0, 4.0, 8.0, 16.0 and 32.0 Mbps were tested. Each test was performed with five 1 min sessions and the results were averaged over all frames over each session. For each speed value, a run using the flow-control algorithm was performed. The round trip delay was defined as the time between sending a signal (i.e. frame is sent to the backend server) and receiving a response (i.e. corresponding prediction is received on edge device)[45].

This study was approved by the University Health Network Research Ethics Board (20-5349). A consent waiver was approved for the use of previously obtained videos for secondary purposes in this research. All raw data consisted of anonymized videos with no personal health information.

Data availability

Annotated datasets used in this study are available upon request through the Global Surgical Artificial Intelligence Collaborative. For access, please email amin.madani@uhn.ca.

Code availability

A demonstration server, model weights, and code are available open source online: https://surg-ai.uhndata.io/.

Received: 9 December 2023; Accepted: 14 August 2024;
Published online: 03 September 2024

References

1. Gawande, A. A., Thomas, E. J., Zinner, M. J. & Brennan, T. A. The incidence and nature of surgical adverse events in Colorado and Utah in 1992. *Surgery* **126**, 66–75 (1999).
2. Rogers, S. O. Jr. et al. Analysis of surgical errors in closed malpractice claims at 4 liability insurers. *Surgery* **140**, 25–33 (2006).
3. Gawande, A. A., Zinner, M. J., Studdert, D. M. & Brennan, T. A. Analysis of errors reported by surgeons at three teaching hospitals. *Surgery* **133**, 614–621 (2003).
4. Wong, E. G., Deckelbaum, D. L. & Razek, T. Global access to surgical care: moving forward. *Lancet Glob. Health* **3**, e298–e299 (2015).
5. Way, L. W. et al. Causes and prevention of laparoscopic bile duct injuries: analysis of 252 cases from a human factors and cognitive psychology perspective. *Ann. Surg.* **237**, 460–469 (2003).
6. Madani, A. et al. What Are the Principles That Guide Behaviors in the Operating Room?: Creating a Framework to Define and Measure Performance. *Ann. Surg.* **265**, 255–267 (2017).
7. Lam, K. et al. Machine learning for technical skill assessment in surgery: a systematic review. *npj Dig. Med* **5**, 1–16 (2022).
8. Pedrett, R., Mascagni, P., Beldi, G., Padoy, N. & Lavanchy, J. L. Technical skill assessment in minimally invasive surgery using artificial intelligence: a systematic review. *Surg. Endosc.* **37**, 7412–7424 (2023).
9. Rueckert, T., Rueckert, D. & Palm, C. Methods and datasets for segmentation of minimally invasive surgical instruments in endoscopic images and videos: a review of the state of the art. *Comput. Biol. Med.* **169**, 107929 (2024).
10. Anteby, R. et al. Deep learning visual analysis in laparoscopic surgery: a systematic review and diagnostic test accuracy meta-analysis. *Surg. Endosc.* **35**, 1521–1533 (2021).
11. den Boer, R. B. et al. Computer-aided anatomy recognition in intrathoracic and -abdominal surgery: a systematic review. *Surg. Endosc.* **36**, 8737–8752 (2022).
12. Madani, A. et al. Artificial intelligence for intraoperative guidance: using semantic segmentation to identify surgical anatomy during laparoscopic cholecystectomy. *Ann. Surg.* **276**, 363–369 (2022).
13. Zhao, H., Jianping, S., Xiaojuan, Q., Wang, X. & Jiaya, J. Pyramid scene parsing network. *ArXiv* https://doi.org/10.48550/arXiv.1612.01105 (2017).
14. He, K., Zhang, X., Ren, S., & Sun, J. Deep residual learning for image recognition. *ArXiv* https://doi.org/10.48550/arXiv.1512.03385 (2016).
15. Owen, D., Grammatikopoulou, M., Luengo, I. & Stoyanov, D. Automated identification of critical structures in laparoscopic cholecystectomy. *Int. J. Comput. Assist. Radiol. Surg.* **17**, 2173–2181 (2022).
16. Owen, D., Grammatikopoulou, M., Luengo, I. & Stoyanov, D. Detection of critical structures in laparoscopic cholecystectomy using label relaxation and self-supervision. 24th International Conference on Medical Image Computing and Computer Assisted Intervention-MICCAI 2021 https://doi.org/10.1007/978-3-030-87202-1_31 (2021).
17. Jaafari, J., Douzi, S., Douzi, K. & Hssina, B. The impact of ensemble learning on surgical tools classification during laparoscopic cholecystectomy. *J. Big Data* **9**, 49 (2022).

18. Twinanda, A. P. et al. EndoNet: A deep architecture for recognition tasks on laparoscopic videos. *IEEE Trans. Med. Imaging* **36**, 86–97 (2017).
19. Coiera, E. The last mile: where artificial intelligence meets reality. *J. Med. Internet Res.* **21**, e16323 (2019).
20. Krizhevsky, A., Sutskever, I. & Hinton, G. E. ImageNet classification with deep convolutional neural networks. *Commun. ACM* **60**, 84–90 (2012).
21. Soper, N. J., Stockmann, P. T., Dunnegan, D. L. & Ashley, S. W. Laparoscopic cholecystectomy. The new 'gold standard'? *Arch. Surg.* **127**, 917–923 (1992).
22. Brunt, L. M. et al. Safe cholecystectomy multi-society practice guideline and state of the art consensus conference on prevention of bile duct injury during cholecystectomy. *Ann. Surg.* **272**, 3–23 (2020).
23. Laplante, S. et al. Validation of an artificial intelligence platform for the guidance of safe laparoscopic cholecystectomy. *Surg. Endosc.* **37**, 2260–2268 (2023).
24. Khalid, M. U. et al. Use of artificial intelligence for decision-support to avoid high-risk behaviors during laparoscopic cholecystectomy. *Surg. Endosc.* **37**, 9467–9475 (2023).
25. Ioffe, S. & Szegedy, C. Batch normalization: accelerating deep network training by reducing internal covariate shift. *ICML'15: Proc. 32nd Int. Conf. Int. Conf. Mach. Learn.* **37**, 448–456 (2015).
26. Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I. & Salakhutdinov, R. R. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv* https://doi.org/10.48550/arXiv.1207.0580 (2012).
27. Ronneberger, O., Fischer, P. & Brox, T. U-Net: convolutional networks for biomedical image segmentation. International Conference on Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015 https://doi.org/10.1007/978-3-319-24574-4_28 (2015).
28. He, K., Zhang, X., Ren, S. & Sun, J. Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification. *arXiv* https://doi.org/10.48550/arXiv.1502.01852 (2015).
29. Buda, M., Saha, A. & Mazurowski, M. A. Association of genomic subtypes of lower-grade gliomas with shape features automatically extracted by a deep learning algorithm. *Comput. Biol. Med.* **109**, 218–225 (2019).
30. Xie, E. et al. SegFormer: simple and efficient design for semantic segmentation with transformers. *arXiv* https://doi.org/10.48550/arXiv.2105.15203 (2021).
31. Wolf, T. et al. HuggingFace's transformers: state-of-the-art natural language processing. *arXiv* https://doi.org/10.48550/arXiv.1910.03771 (2019).
32. Amann, J. et al. Explainability for artificial intelligence in healthcare: a multidisciplinary perspective. *BMC Med. Inform. Decis. Mak.* **20**, 310 (2020).
33. Reddy, S. Explainability and artificial intelligence in medicine. *Lancet Digit. Health* **4**, e214–e215 (2022).
34. Bertsekas, D. & Gallager, R. Data networks (2nd edition). Chapter 6 flow control 492-536 (Athena Scientific, 2021).
35. Malik, P., Pathania, M. & Rathaur, V. K. Overview of artificial intelligence in medicine. *J. Fam. Med Prim. Care.* **8**, 2328–2331 (2019).
36. Maier-Hein, L. et al. Metrics reloaded: recommendations for image analysis validation. *Nat. Methods* **21**, 195–212 (2024).
37. Madani, A. et al. Measuring intra-operative decision-making during laparoscopic cholecystectomy: validity evidence for a novel interactive web-based assessment tool. *Surg. Endosc.* **31**, 1203–1212 (2017).
38. Madani, A. et al. Measuring decision-making during thyroidectomy: validity evidence for a web-based assessment tool. *World J. Surg.* **42**, 376–383 (2018).
39. Madani, A., Grover, K. & Watanabe, Y. Measuring and teaching intraoperative decision-making using the visual concordance test: deliberate practice of advanced cognitive skills. *JAMA Surg.* **155**, 78–79 (2020).
40. Horn, B. K. & Schunck, B. G. Determining optical flow. *Artif. Intell.* **17**, 185–203 (1981).
41. Hameed, M. S. et al. What is the educational value and clinical utility of artificial intelligence for intraoperative and postoperative video analysis? A survey of surgeons and trainees. *Surg. Endosc.* **37**, 9453–9460 (2023).
42. Lou, S. et al. Artificial intelligence for colorectal neoplasia detection during colonoscopy: a systematic review and meta-analysis of randomized clinical trials. *EClinicalMedicine* **66**, 102341 (2023).
43. Gimeno-García, A. Z., Hernández-Pérez, A., Nicolás-Pérez, D. & Hernández-Guerra, M. Artificial intelligence applied to colonoscopy: is it time to take a step forward? *Cancers* **15**, 2193 (2023).
44. Rauch, G. Socket. IO: the cross-browser WebSocket for realtime apps. http://socket.io/ 2012.
45. Chen, X. et al. Measuring TCP round-trip time in the data plane. SPIN '20: Proceedings of the Workshop on Secure Programmable Network Infrastructure. https://doi.org/10.1145/3405669.3405823 (2020).

Acknowledgements

This study was funded by the University Health Network Foundation. The funder played no role in study design, data collection, analysis and interpretation of data, or the writing of this manuscript. MB holds a CIFAR CCAI Chair.

Author contributions

Conceptualization, supervision: A.M., M.B., P.M. Methodology: S.P., J.H. Formal analysis: S.P., J.H. Data verification, validation and visualization: S.P., J.H., H.Z., P.M. Software development: H.Z., S.P. Data curation: C.M., A.M. Writing-original draft: S.P., J.H., H.Z., P.M. Writing-review and editing: A.M., M.B., S.P., P.M., C.M. All authors have read and agreed to the published version of the manuscript.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s41746-024-01225-2.

Correspondence and requests for materials should be addressed to Michael Brudno or Amin Madani.