Medical image video recorder with computer vision and face blurring

Gravador de vídeo de imagens médicas com visão computacional e desfoque facial

Grabador de vídeo de imágenes médicas con visión por computadora y desenfoque facial

Eduardo Mobilon[1], Igor Marques de Araujo[1], Luiz Antonio Buschetto Macarini[1], Luiz Eduardo Pita Mercês Almeida[1], Rodrigo Bernardo[1], Luis Paulo Fernandes de Barros[1], Renata Bastianon[1], and Ricardo Mendes Alves Pereira[2]

1 Diretoria de Tecnologia e Inovação, CPQD, Campinas (SP), Brasil
2 Instituto de Cirurgia Ginecológica Ricardo Pereira, São Paulo (SP), Brasil

Autor correspondente: Dr. Eduardo Mobilon
E-mail: mobilon@cpqd.com.br

Abstract

Modern solutions for recording medical procedures represent cutting-edge technology that is still emerging and facing challenges. This paper presents the Life Surgery Box, a Brazilian standalone multi-modal and synchronized image video recorder. Objective: presenting the development and prototyping of the equipment, intended for use in both operating rooms and medical offices. Method: involves the description of its hardware and software architectures, with a focus on an artificial intelligence-based face-blurring algorithm. Results: highlight the performance optimizations for efficient video processing and the artifacts generated by the equipment. Conclusion: the proposed solution exemplifies technological advancements and stands as an innovative contribution to healthcare technology.

Keywords: Laparoscopic Surgery; Video Recording; Artificial Intelligence

Resumo

Soluções modernas para registro de procedimentos médicos representam tecnologia de ponta que ainda está surgindo e enfrentando desafios. Este artigo apresenta o Life Surgery Box, um gravador de vídeo brasileiro autônomo de imagens sincronizadas e multimodais. Objetivo: apresentar o desenvolvimento e prototipagem do equipamento, destinado uso tanto em salas cirúrgicas quanto em consultórios médicos. Método: envolve a descrição de suas arquiteturas de hardware e software, com foco em um algoritmo de desfoque facial baseado em inteligência artificial. Resultados: destacam as otimizações de desempenho para processamento eficiente de vídeo e os artefatos gerados pelo equipamento. Conclusão: a solução proposta exemplifica os avanços tecnológicos e representa uma contribuição inovadora para a tecnologia em saúde.

Descritores: Cirurgia Laparoscópica; Gravação de Vídeo; Inteligência Artificial

Resumen

Las soluciones modernas para registrar procedimientos médicos representan una tecnología de vanguardia que aún está surgiendo y enfrentando desafíos. Este artículo presenta Life Surgery Box, un videograbador brasileño autónomo de imágenes sincronizadas y multimodales. Objetivo: presentar el desarrollo y prototipado del equipo, destinado a ser utilizado tanto en quirófanos como en consultorios médicos. Método: consiste en describir sus arquitecturas de hardware y software, centrándose en un algoritmo de desenfoque facial basado en inteligencia artificial. Resultados: destacan las optimizaciones de rendimiento para el procesamiento eficiente de video y los artefactos generados por el equipo. Conclusión: la solución propuesta ejemplifica los avances tecnológicos y representa un aporte innovador a la tecnología de la salud.

Descriptores: Cirugía Laparoscópica; Grabación de Vídeo; Inteligencia Artificial

Introduction

Laparoscopic surgery has transformed surgical practices by providing less invasive options compared to traditional open surgeries, resulting in reduced patient discomfort and faster recovery times. The use of video recording in this scenario is invaluable for medical education, training, and documentation [(1)]. It not only spreads knowledge among healthcare professionals but also advances surgical techniques, enhances patient outcomes, promotes transparency in the medical field, and encourages collaborative learning and continuous improvement in minimally invasive surgery [(2–4)].

Analyzing successful operations helps identify various error sequences, aiding surgeons in case preparation, error prevention, and mitigating their consequences [(2)].

The operating room (OR) is a dynamic and intricate interprofessional environment where various factors like distractions, skills, communication, and equipment issues can affect intraoperative procedures and ultimately patient safety (5).

A study assessed how video recording affects colonoscopy quality, recognizing the procedure's operator-dependent nature (6). Medical professionals performed routine colonoscopies with and without prior video recording knowledge. Results showed a 49% increase in inspection time and improved mucosal inspection technique after awareness of recording, indicating immediate physician performance enhancement.

Another study (7) examined how audio-video recording in the OR affects the focus of both the surgeon and his assistant during laparoscopic surgeries. The results indicated that recording reduces unnecessary conversation time and has the potential to improve intraoperative safety and surgical outcomes.

Modern solutions for recording medical procedures and consolidating real-time data from the operating room represent cutting-edge technology that is still emerging and facing challenges (5). Despite many opportunities for improving quality and exploring new applications, concerns about patient and staff privacy may limit routine use. Potential solutions include creating deidentified videos that retain enough data for their intended purposes (8).

Synchronization and image compositing during surgical event recording provide significant benefits, improving documentation quality, analysis, and communication in medicine. Aligning various imaging sources like endoscopic views, OR views, and patient vitals into a synchronized composite image offers a comprehensive view of the surgical procedure. This detailed documentation is valuable for legal and educational purposes, offering an accurate and chronological record of the surgical intervention.

This paper presents the development and prototyping of a standalone equipment called Life Surgery Box, designed as a multi-modal and synchronized image video recorder intended for use in both operating rooms and medical offices. The subsequent sections detail the hardware and software architectures, with a focus on AI-driven computer vision and face-blurring solutions.

Methods

Conceived by RILAP and developed by CPQD Telecom R&D Center, the Life Surgery Box is a Brazilian technology apparatus that provides independent input video, audio, and data signal interfaces with integrated electrical isolation to be connected to laparoscopic cameras, multiparameter monitors, bispectral index (BIS) monitors, and operating room cameras. Figure 1 shows pictures of the equipment (a) and the complete solution mounted in a mobility rack (b).

Independent signal sources are synchronously combined into a composite image that is then recorded (with ambient audio) as a 1080p H.264 compressed video file. Recorded events are securely stored using strong 256-bit advanced encryption standard (AES) cryptography and can be promptly uploaded to a cloud storage and backup environment upon request. A touchscreen video monitor enables the user/operator to control the equipment through a graphical user interface (GUI) and the integrated software system also incorporates a database event manager for file retrieval, copy, and playback.

To address privacy concerns, video recordings can also be post-processed for deidentification using embedded artificial intelligence (AI) to detect and blur human faces in the OR camera view.

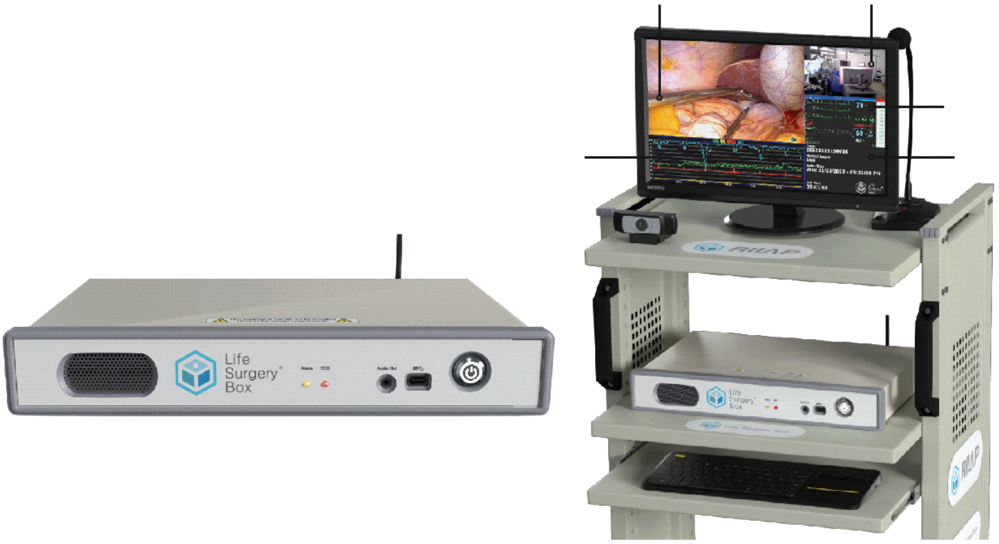


Figure 1 – Life Surgery Box equipment (a) and the complete solution mounted in a mobility rack (b).

Hardware Architecture and Design

Life Surgery Box electronic hardware architecture block diagram in shown in Figure 2. The processing unit is based on a Jetson TX2 single board computer (SBC) from Nvidia, which integrates a dual-core 64-Bit CPU, a quad-core ARM Cortex-A57 MPCore, and a 256-core GPU. Dedicated image processing and H.264 video codec (coder/decoder) engines provide the necessary hardware acceleration for real-time video capture, compositing, encoding, and recording.

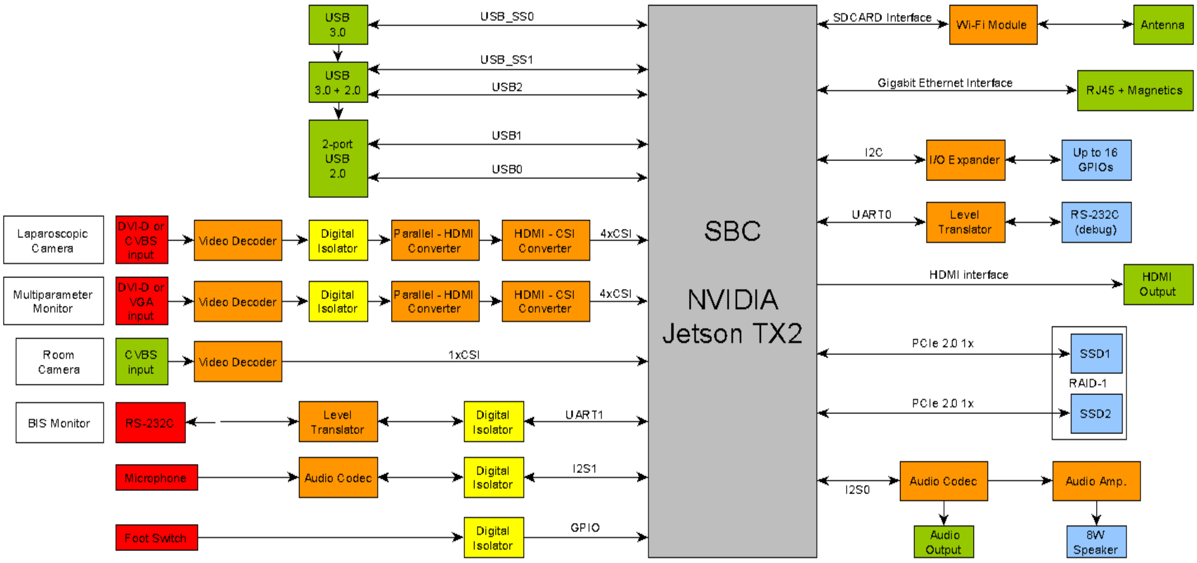


Figure 2 – Life Surgery Box electronic hardware architecture block diagram.

The main video signal source interfaces are the laparoscopic camera, multiparameter monitor, and the room camera. BIS monitor data are received by a standard RS-232C serial interface. Audio is captured by a room microphone and image screenshots can be taken by a foot switch. All these main input interfaces are electrically isolated by a galvanic barrier, providing the means of patient protection (MOPP) required by IEC 60.601-1 standard [(9)].

Additional available interfaces include a set of USB ports, Gigabit Ethernet, Wi-Fi, HDMI output for an external video monitor, and audio outputs (line and integrated speaker).

Data are stored in self-encrypting drives (SEDs), with two 1 TB solid state drives (SSDs) connected to the SBC by a single-lane peripheral component interconnect express (PCIe) interface. A redundant array of independent disks (RAID) level 1 architecture was then implemented on the processor operating system (OS) to provide disk mirroring, mitigating the risk of data loss.

All signal interfaces directly connected to electromedical equipment or to other devices within the so-called patient environment (such as the microphone or foot switch) are electrically isolated. The necessary galvanic barrier was implemented by capacitive digital isolators with a 5 kV dielectric strength.

Software Architecture

As depicted in Figure 3, the Life Surgery Box software system was conceived as a three-component architecture — the graphical user interface (GUI), the system controller, and the video application, all built upon a Ubuntu-based OS (an Nvidia customized Linux distribution).

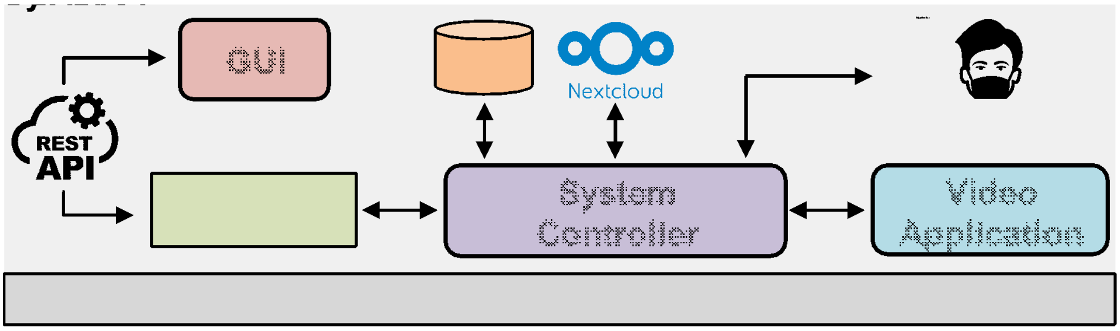


Figure 3 – Software architecture block diagram with the main components.

The GUI was designed as a Web application, implemented using Node.js and React and opened from an embedded Flask Web server by a Chromium browser running in kiosk mode. The system controller module is primarily made up of Python scripts that interact with the other software components, an SQLite database (DB), and a Nextcloud content collaboration and storage platform, managing the equipment functionality and starting other software operations. Finally the video application, coded in C language and based on the GStreamer open source multimedia framework, handles video and audio capture, image compositing, H.264 hardware accelerated encoding, and recording. Device drivers were also developed and/or adapted for the OS to access and control hardware components.

A user management system ensures authentication, granting access and control to recorded events in accordance with a profile-defined policy.

Recorded events can undergo post-processing using AI to achieve deidentification in the OR camera view. As illustrated in Figure 4, human faces are detected by a computer vision-based model before being anonymized by a blurring algorithm, which applies a weighted average over a neighborhood of pixels around each pixel in the image, smoothing color transitions and image details, thus creating the desired blur effect.

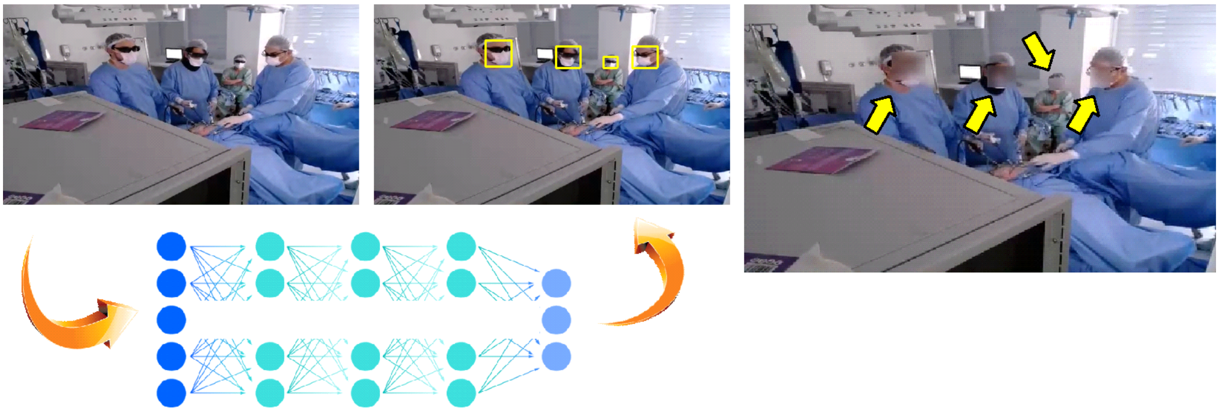


Figure 4 – AI-based video post-processing with face detection and blurring.

Computer Vision – Strategy and Available Models

Face detection is a crucial task in Machine Learning (ML), essential for various applications like facial recognition, security, and video analysis. Despite significant progress driven by improved ML algorithms and enriched datasets, challenges such as lighting variations, diverse poses, facial expressions, scale differences, and occlusions (e.g., glasses and masks) underscore the ongoing complexity of the task.

For surgical environments, like those captured by the Life Surgery Box equipment, a computer vision-based face detection solution was developed with the following approach:

- Study and selection of an initial detection model;
- Annotation of a dataset specific to the problem;
- Creation of an experimental protocol;
- Model training for face detection with surgical environments' specificities.

With a plethora of models already researched and published in the realm of face detection, the selected approach was to delve into surveys and reviews comparing them. In this work, the two most influential articles available at the time were meticulously selected for thorough examination [10–11], both aiming to describe and compare ML techniques and models focused on face detection.

Other important factors were considered when choosing the base model, including:

- Robustness: the model was expected to have been pre-trained on a diverse database, so that it could easily adapt to specific use cases;
- Time Efficiency: while real-time inferences were not necessary, it was crucial for the system to maintain moderate efficiency to prevent time constraints from becoming prohibitive;
- Size: to be embedded in a system with low memory;
- Availability: given that the model would specialize in a new dataset, open-source code or accessible neural network weights were essential features.

Considering all these aspects, the selected model was RetinaFace [12, 13], which is very robust since it was pre-trained on the WIDER FACE dataset [14], known for having high data variability, containing faces of different people, in different environments, with varied occlusions, lighting, and scales. Moreover, the model presents an option with a relatively compact central structure — MobileNet-0.25 [15], ensuring swift inference times and minimal memory usage.

Alternative models [16–19] were assessed, but when compared with RetinaFace they were deemed less suitable for the unique requirements of the Life Surgery Box solution.

Computer Vision – Dataset, Annotation, and Performance Metric Selection

The primary dataset used for model training comprised over 60 video recordings of laparoscopic surgeries, captured by the Life Surgery Box equipment. Due to the diverse nature of the provided video files, an evaluation was conducted to ensure their validity and feasibility. Subsequently, video segments were extracted from these surgical procedures, encompassing a range of lighting conditions, colors, camera angles, number of participants, and scale, given their typically extended duration lasting hours.

Data annotation was then performed using the CVAT (Computer Vision Annotation Tool) platform. Faces in each frame of all video segments were labeled by drawing rectangles around them. A total of 57 one-minute video snippets, comprising 1800 frames at 30 fps, were annotated for model training.

There are established and widely recognized metrics to assess the performance of models in face detection. The main problem consists of outlining rectangular quadrants around human faces on video frames under analysis. Through the comparison of two rectangles — one generated by the model and the other representing the ground truth — the Intersection over Union (IoU) metric can be employed to determine the accuracy of the predicted bounding box and assess its validity as a detection. Following the analysis of both errors and successes, classic metrics such as precision, recall, and others can be leveraged to evaluate the performance of the model. Mean Average Precision (mAP), a widely adopted metric for evaluating object detection systems, was selected for this work, offering a comprehensive and robust assessment of the model's performance.

Computer Vision – Experimental Protocol and Model Training

The training of the RetinaFace model followed a conventional methodology, involving the standard division of the dataset into three subsets: training, validation, and testing. The training process involves using the training set to train the model, while simultaneously assessing the model's performance using the validation set.

Out of the 57 video segments, 31 were selected for training, 12 for validation, and 14 for testing. Given the dataset's relatively small size, a strategic methodology was employed to ensure a balanced distribution of data across all sets. This involved keeping the data distribution similar between them, considering key factors pertinent to the task, which varied among different video recordings. These factors included the average number of individuals present in the operating room, lighting conditions, the visibility of the patient in a particular camera angle, the color of surgical team attire, and facial scale. Other variables, such as the types of personal protective equipment worn by the teams, remained consistent and did not require individual evaluation in each video segment.

After constructing and partitioning the dataset, multiple experiments were conducted involving variations of hyperparameters to identify the optimal combination for model training. The search for these hyperparameters predominantly centered on the arguments of the Stochastic Gradient Descent (SGD) optimization function within the PyTorch framework library. They include the Learning Rate, Momentum, and Weight Decay. The best-performing values were determined as Learning Rate = 0.001, Momentum = 0.95, and Weight Decay = 0.05. Moreover, the initial learning rate was reduced by 10% after the model had been trained on 1500 input examples, and by an additional 1% after reaching 2750 input examples.

Upon the conclusion of the training phase, the final model underwent rigorous evaluation using the test set to ascertain its overall efficacy and generalization capabilities using the mAP metric.

Computer Vision – AI Code and Optimizations

The first version of the AI code was developed in Python, using the PyTorch framework for the neural network implementation. Due to limitations on the availability and possibility of the OS Kernel updates, the toolset used was Python 3.6, PyTorch 1.4, and Torchvision 0.5.0.

Although face detection and blurring are executed in a post-processing phase, the initial processing time requirement was 1x or, in other words, the same duration of the original recorded video. However, initial tests showed this time to be 3x, indicating the need for a code optimization. To identify possible bottlenecks, the AI code was divided into ten sections, described in Table 1, each being measured to evaluate the corresponding execution time.

The first bottleneck was found in step 3, with these array operations being quite costly to perform in Python. To mitigate this, Cython was used to enable the creation of Python extensions in C, thereby achieving performance levels closer to that of native codes in this language. The second limitation was the process of saving the final frames to disk. Initially, the FFmpeg tool was used to read frames from the original video file and, after the face detection and blurring, save them to a buffer for generating the post-processed video file. Specifically, the frame-saving process accounted for about 70% of the total processing time. This was strongly optimized by the use of GStreamer — the same open source multimedia framework used in the main Video Application.

Table 1 – Ten sections of the AI source code with their corresponding descriptions.

Step	Section	Description

1	Time before the main loop	Time measured to perform tasks before the image processing loop (e.g., loading the model into memory)
2	Crop image	Crop the area of interest (OR camera view)
3	Image processing	Image conversion to *float32*, normalization, RGB channel transposition, conversion to Numpy Array, and sending to the GPU
4	Inference	Performing inference with the neural network
5	Variance decoding	Conversion of tensors with scores to Numpy Arrays
6	Index (score) filtering	Removing indices (detections) below the minimum confidence index
7	Index (score) sorting	Sorting detections from highest to lowest indices
8	NMS application	Non-Max Suppression application
9	Blur application	Applying blur to the detected faces
10	Saving frame to disk	Insertion of the frame into the buffer for generating the complete video at the end of the process

Given the considerably faster speed of reading video frames compared to processing them, recurrent memory overflows were observed during video processing. To mitigate this issue, a queue with a capacity of 500 items was implemented to control frame retrieval. Once it reached maximum capacity, the retrieval of new frames was temporarily halted. As frames were processed and the queue emptied, additional frames were fetched from the video source.

To speed up the process, the hardware codec available in the Jetson TX2 SBC was used by the GStreamer components nvv4l2decoder, nvv4l2h264enc, and nvvideoconvert. It efficiently encodes and decodes raw video data into/from a compressed format.

Four tests were conducted to evaluate the performance improvements:

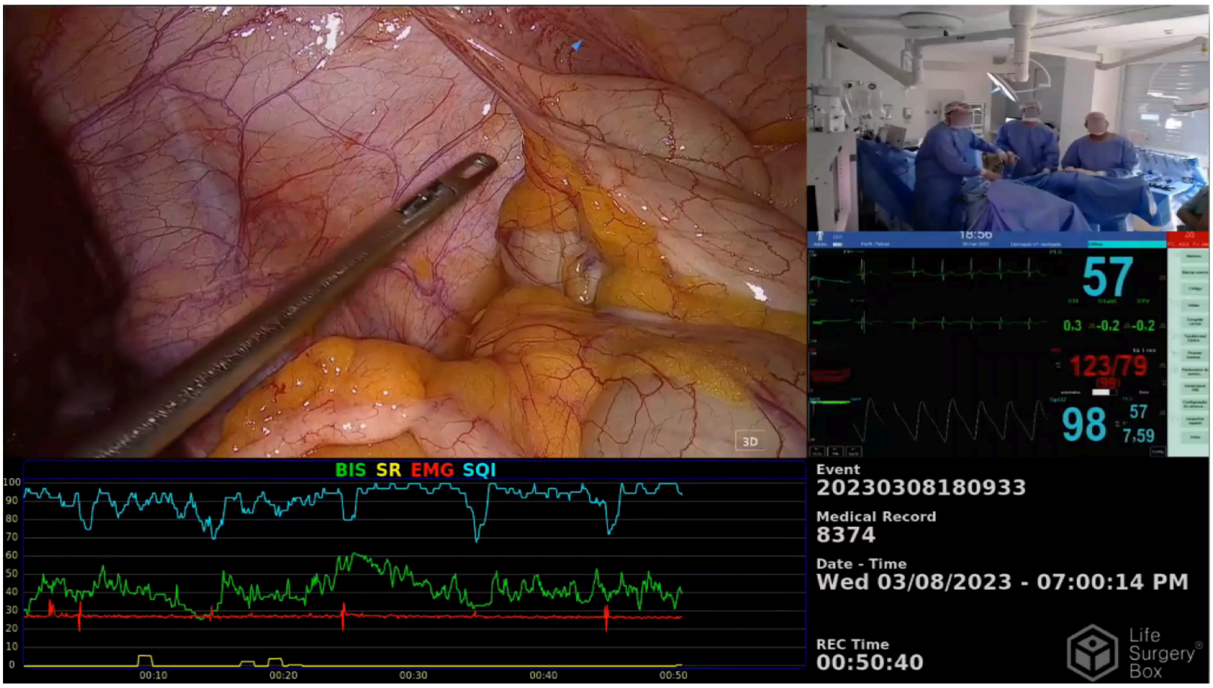
- Test 1: Using GStreamer instead of FFmpeg;
- Test 2: Test 1 scenario with the addition of the TX2 HW decoder;
- Test 3: Test 2 scenario with the addition of the TX2 HW encoder;
- Test 4: Test 3 scenario with the addition of Cython and increasing the Batch Size from 4 to 16.

These tests were performed with a short source video file lasting 182 seconds. Its main parameters are the resolution of 1920x1080, H.264 encoding (Constrained Baseline Profile), 30 fps, and a bitrate of 6992 kbps. The audio was encoded as MPEG-4 AAC, Mono, with a sampling rate of 44100 Hz and a bitrate of 127 kbps.

Results

Figure 5 presents a screenshot of the 1080p video recorded by the equipment, showing the generated composite image with the endoscopic view, OR view (with face blurring), vitals, BIS, and event data.

A portable document format (PDF) file is also created for each recorded event, containing additional information including event identification, start/finish recording time, place, medical record, medical procedure, and patient/physician names.



Event
20230308180933

Medical Record
8374

Date - Time
Wed 03/08/2023 - 07:00:14 PM

REC Time
00:50:40

Figure 5 – Screenshot of the 1080p video-recorded composite image with the endoscopic view, OR view (with face blurring), vitals, BIS, and event data.

Regarding the performance of the computer vision model, Table 2 shows the mean Average Precision (mAP) of the original and fine-tuned RetinaFace models. This metric was computed individually for every frame within the test set and then averaged. While the training process led to a remarkable improvement exceeding 394%, it is noteworthy that the performance did not approach the optimal value of one, indicating the inherently challenging nature of the use case. This underscores the success of the work while also highlighting opportunities for further enhancement in future endeavors.

Table 2 – Performances of the original and fine-tuned RetinaFace models.

Model	mAP (test set)
RetinaFace (original)	0.069
RetinaFace (fine-tuned in the dataset)	0.272

In terms of AI code optimization, Table 3 presents the test results for each of the ten code sections, with time measurements conducted using Python's time library. The total processing time achieved is approximately 1.2 times the duration of the original video file (220 seconds over 182 seconds).

Table 3 – Processing time (in seconds) obtained in each section in the four tests performed.

Step	Test 1	Test 2	Test 3	Test 4
1	0.014758890	0.0219874596	0.01590616246	0.02340747885
2	0.000426454	0.000447275495	0.000505133137	0.000631802246
3	0.195379955	0.2072014936	0.2185739399	0.3002965875
4	0.372694030	0.3274036893	0.387110648	0.1801836878
5	0.064589400	0.06971430126	0.06264788726	0.1797695116
6	0.004027067	0.004590305818	0.004044405445	0.005564976166
7	0.000799248	0.000914543702	0.000778997076	0.001024662513
8	0.015306156	0.01657104441	0.01396160592	0.01896454959
9	0.237992105	0.2609213749	0.2162554348	0.1693081815
10	0.094026695	0.09024851191	0.08021578599	0.1208485623
Total	**333.15 sec**	**305.08 sec**	**290.21 sec**	**220 sec**

After two prototyping cycles, a fully functional electronic hardware was available and compliant with the necessary certification tests, such as electromagnetic compatibility and interference (EMC/EMI), surge immunity, flicker, and electrostatic discharge (ESD). High potential (HiPOT) tests also confirmed the effectiveness of the means of patient protection (MOPP) insulation, limiting leakage currents to less than 300 μA as required by IEC 60.601-1. A pilot production batch of ten units was manufactured, facilitating the showcasing of the Life Surgery Box solution at medical exhibitions in North/South America and Europe.

Conclusion

Video recording during laparoscopic procedures enhances medical practice, aiding education, error identification, and patient safety. The development of the Life

Surgery Box solution described in this paper exemplifies technological advancements, incorporating AI-driven computer vision solutions for detecting and blurring human faces to ensure privacy. Results highlighted the algorithm optimizations, a screenshot of the 1080p video-recorded composite image with the expected anonymization, and a pilot production batch manufacturing for worldwide medical exhibitions.

References

1. Scherer L A, Chang M C, Meredith J W, Battistella F D. Videotape review leads to rapid and sustained learning. Am J Surg. 2003;185(6):516–20. https://doi.org/10.1016/S0002-9610(03)00062-X.

2. Bonrath E M, Gordon L E, Grantcharov T P. Characterising 'near miss' events in complex laparoscopic surgery through video analysis. BMJ Qual Saf. 2015; 24(8):516–21. https://doi.org/10.1136/bmjqs-2014-003816.

3. Hu Y Y, Peyre S E, Arriaga A F, et al. Postgame analysis: using video-based coaching for continuous professional development. J Am Coll Surg. 2012; 214(1):115–24. https://doi.org/10.1016/j.jamcollsurg.2011.10.009.

4. Bogen E M, Augestad K M, Patel H R, Lindsetmo R O. Telementoring in education of laparoscopic surgeons: An emerging technology. World J Gastrointest Endosc. 2014;6(5):148–55. https://doi.org/10.4253/wjge.v6.i5.148.

5. Møller K E, Sørensen J L, Topperzer M K, Koerner C, Ottesen B, Rosendahl M, Grantcharov T, Strandbygaard J. Implementation of an Innovative Technology Called the OR Black Box: A Feasibility Study. Surg Innov. 2023;30(1):64-72. https://doi.org/10.1177/15533506221106258.

6. Rex D K, Hewett D G, Raghavendra M, Chalasani N. The impact of videorecording on the quality of colonoscopy performance: a pilot study. Am J Gastroenterol. 2010;105(11):2312-7. https://doi.org/10.1038/ajg.2010.245.

7. Bergström H, Larsson L G, Stenberg E. Audio-video recording during laparoscopic surgery reduces irrelevant conversation between surgeons: a cohort study. BMC Surg. 2018;18(1):92. https://doi.org/10.1186/s12893-018-0428-x.

8. Silas M R, Grassia P, Langerman A. Video recording of the operating room--is anonymity possible? J Surg Res. 2015;197(2):272-6. https://doi.org/10.1016/j.jss.2015.03.097.

9. International Electrotechnical Commission. IEC 60601-1: Medical electrical equipment - Part 1: General requirements for basic safety and essential performance. Revision 3.2. August 2020.

10. Minaee S, Luo P, Lin Z, Bowyer K. Going Deeper Into Face Detection: A Survey. 2021. ArXiv. /abs/2103.14983.

11. Feng Y, Yu S, Peng H, Li Y, Zhang J. Detect Faces Efficiently: A Survey and Evaluations. 2021. ArXiv. https://doi.org/10.1109/TBIOM.2021.3120412.

12. Deng J, Guo J, Zhou Y, Yu J, Kotsia I, Zafeiriou S. RetinaFace: Single-stage Dense Face Localisation in the Wild. 2019. ArXiv. /abs/1905.00641.

13. RetinaFace in PyTorch. https://github.com/biubug6/Pytorch_Retinaface.

14. Yang S, Luo P, Loy C C, Tang X. WIDER FACE: A Face Detection Benchmark. 2015. ArXiv. /abs/1511.06523.

15. Howard A G, Zhu M, Chen B, Kalenichenko D, Wang W, Weyand T, Andreetto M, Adam H. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. 2017. ArXiv. /abs/1704.04861.

16. Zhu Y, Cai H, Zhang S, Wang C, Xiong Y. TinaFace: Strong but Simple Baseline for Face Detection. 2020. ArXiv. /abs/2011.13183.

17. Liu Y, Tang X, Wu X, Han J, Liu J, Ding E. HAMBox: Delving into Online High-quality Anchors Mining for Detecting Outer Faces. 2019. ArXiv. /abs/1912.09231.

18. Li J, Wang Y, Wang C, Tai Y, Qian J, Yang J, Wang C, Li J, Huang F. DSFD: Dual Shot Face Detector. 2018. ArXiv. /abs/1810.10220.

19. He K, Zhang X, Ren S, Sun J. Deep Residual Learning for Image Recognition. 2015. ArXiv. /abs/1512.03385.