

# Ushering in the next generation of autonomous surgical robots? current trends and future possibilities with data driven physics simulation and domain randomization

Youssef Ghosn<sup>a#</sup>, Mohammed Hussein Kamareddine<sup>b#</sup>, Geroge salloum<sup>c</sup>, Elie Najem<sup>a</sup>, Ayman Ghosn<sup>d</sup>, Charbel Saade<sup>e\*</sup>

<sup>a</sup> Diagnostic Radiology Department, American University of Beirut Medical Center, Beirut, Lebanon.

<sup>b</sup> Faculty of Medicine and Health, School of Public Health, University of Sydney, Sydney, Australia.

<sup>c</sup> Department of Computer Engineering, Lebanese American University, Beirut, Lebanon.

<sup>d</sup> XOL Automation S.A.L, Beirut, Lebanon.

<sup>e</sup> Faculty of Health Sciences, American University of Beirut Medical Center, Beirut, Lebanon.

## Abstract

As artificial intelligence (AI) plays an ever-increasing role in medicine, various designs that implement machine learning (ML) are being introduced in an effort to develop surgical robots that perform a variety of surgical techniques without human interference. However, current attempts in creating autonomous surgical robots (ASRs) are hindered by the amount of time needed to train a robot on a physical sett, the incredible amount of physical and/or synthetic (artificial) data needed to be collected and labeled, as well as the unaccountable and unpredictable characteristics of reality. Progress outside of the medical field is being made to address the general limitations in autonomous robotics.

Herein, we present a review of the basics of machine learning before going through the current attempts in creating ASRs and the limitations of current technologies. Finally, we present suggested solutions for these limitations, mainly data driven physics simulations and domain randomization, in an attempt to create a virtual training environment as faithful to and as random as the real world that could be transferred to a physical setting. The solutions suggested here are based on techniques incorporated and strides being made outside of the medical field that could usher in the next generation of autonomous surgical robotics designs.

**Keywords:** Autonomous surgical robots; robotic surgery; artificial intelligence; machine learning; physics simulation; domain randomization.

## INTRODUCTION

Technological advancements in hardware and software increasingly play an imperative role in the evolution of contemporary medical/surgical techniques and paradigms. This, in addition to the high liability of medical

errors, an increased workload, coupled with a reduced workforce and an aging population, are incentivizing experts to acquaint themselves with computerized assistants or to introduce certain automated surgical interventions [1-3]. One such technology is artificial intelligence (AI), specifically machine learning (ML). This article reviews the current role of ML techniques in surgery with a focus on autonomous robotics surgery (ARS). Also, we provide a perspective on future possibilities that could help in enhancing the effectiveness of autonomous surgical robots (ASRs), mainly data driven physics simulation and domain randomization. The use of ML in electronic medical record systems, diagnostics and medical imaging is out of the scope of this review. Searches were performed on Google scholar, Medline,

# These authors contributed equally to the manuscript.

\* Corresponding author: Charbel Saade, PhD

Mailing address: Faculty of Health Sciences, American University of Beirut Medical Center, Beirut, Lebanon. P.O.Box: 11-0236 Riad El-Solh, Beirut 1107 2020, Lebanon.

E-mail: mdct.com.au@gmail.com

Received: 20 May 2020 Accepted: 17 June 2020

PubMed, Scopus, Cochrane library and IEEE using various combinations of keywords: Autonomous surgical robots, surgical and medical robotics, artificial intelligence, machine learning, physics simulation and domain randomization.

As AI systems are being continuously adopted in medicine, there has been increasing interest in “autonomous” surgical robots that can assist surgeons or even perform portions of an intervention independent of human guidance or control<sup>[3,4]</sup>. An autonomous intelligent robot can be achieved using different variations of AI. ML is a subset of AI and an increasingly growing field. It is popular as it permits efficient processing of large quantities of data for analysis, interpretation and decision making while providing computers with the ability to learn and perform a range of tasks without being explicitly programmed to do so. Already widely used in electronic medical record systems, medical imaging and diagnostics, it is expected that ML will play a pivotal role in surgical and interventional procedures<sup>[1,3,5]</sup>.

ML agents can acquire surgical skills in a variety of ways, one of which is, for example, through demonstration by human experts<sup>[3]</sup>. Currently, intelligent surgical robots with varying degrees of autonomy are proving to be comparable to surgeons at some tasks, such as suturing, locating wounds and tumor removal. These intelligent surgical assistants could surpass the current state of the art commercial surgical robots and promise good results and a wider access to specialized procedures<sup>[1,3,5]</sup>.

As promising as this might seem, debilitating limitations currently hinder substantial progress in medical application of AI generally, and ASRs specifically. Mostly, these limitations are linked to the current available AI technologies and partially to some unique characteristic of AI application in medicine. The main limitations include: The need for high-quality medical/surgical data which slows the process of developing effective agents while requiring large scale collaborative efforts- a modeling challenge that hinders our ability to accurately “model” a surgical environment that replicates the dynamic and deforming nature of the living body- and the inability of intelligent agents in general and surgical robots specifically to adapt to unknown or yet unobserved situations<sup>[3]</sup>.

Interestingly, new technological advancements in AI software designs are being currently developed that, we think, could help us overcome the aforementioned limitations. Two particular new AI advancements that could be of use are data driven physics simulation environments and domain randomization. First, we will go through a general overview on ML parameters implemented in current autonomous surgical robots while exploring some examples of present automated surgical robotic technologies, we will then discuss some of the

limitations of the current technologies and go through our proposed solutions to overcome current designs drawbacks.

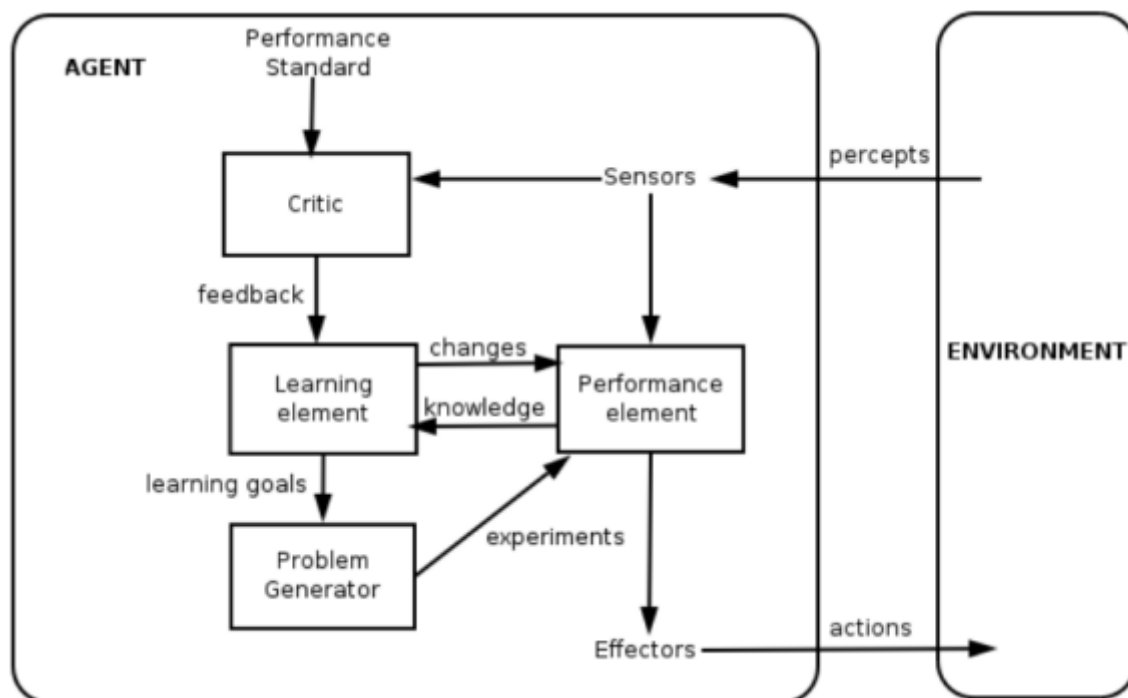
## DESIGN

A robot is a system that has three main components: a set of sensors that detect the robot’s environment, actuators (or end effectors) that interact with and within the environment and a control architecture that processes sensory data and generates actions<sup>[3]</sup>(Figure 1). ML is mainly involved in the control architecture, enabling the robot to “understand” the sensory input and generate a proper action. In order to do so, the agent must learn to generate a certain action in the context of a set of sensory inputs and desired goals. So, how can a robot learn a surgical skill? First, it could learn from human demonstration by observing experiments conducted by trained experts. The robot could also learn from its own interaction with the environment, by evaluating the appropriateness of its own actions to reach certain target states or goals<sup>[3,6]</sup>. In order to understand current autonomous robotic surgery technologies and future perspectives, a general knowledge of the main subsets of machine learning is needed. The three main categories of machine learning are supervised learning (SL), unsupervised learning (UL) and reinforcement learning (RL)<sup>[7]</sup>. Many agents use variations and combinations of these three categories.

In SL, training data are considered “labeled”, ie, the data consist of a set of known input vectors along with a set of known matching target vectors. The software creates a function that links the input object with the corresponding output<sup>[8]</sup>. For example, consider we have a set of photos of thousands of lung pathologies (pneumothorax, pneumonia etcetera). The data is considered labeled if each photo identifies certain features of the radiograph such as opacification values (input vectors) and each type of pathologies (target vector). Eventually, SL seeks to build a predictor model that predicts target vectors for new input vectors. Learning consists of finding optimal parameter values for the predictor model<sup>[3]</sup>.

In UL, the training data is considered unlabeled and consists of a set of input vectors without their corresponding target vectors. UL aims to discover correlations and structure in the data. Whether using UL or SL, Gaussian Mixture Model (GMM) and Gaussian Mixture Regression (GMR) based learning could be added to fine tune the learning process and get more reliable demonstration<sup>[3,9]</sup>.

Reinforcement learning (RL) is concerned with how intelligent agents ought to take actions in an environment in order to maximize cumulative reward. The difference between RL and SL is that RL does not need labelled input/output data pairs, the training data is mostly generated through direct interaction with the environment, and RL does not need explicit correction of sub-optimal actions.



**Figure 1.** A general learning agent consists of sensors, effectors and a control architecture. The control architecture has a “learning element” that uses feedback from the “critic” on the quality of tasks execution and determines how the performance should be modified. The control architecture also has a problem generator that suggests actions that could lead to new and informative experiences.

RL focuses on exploring the environment (states) in a trial and error approach in order to create and adjust a policy that permits the agent to perform a certain task (action). RL’s end goal is to learn a policy that represents a mapping from states to actions. For example, suppose that we want to train a robot to tie a knot. The agent will explore the environment, gradually develop a certain policy (creating and adjusting a suturing technique) in order to be eventually capable of performing the task of tying the knot. A value of a state-action pair is created and represents how good it is for the robot to perform an action in a given condition or state [3,7].

RL can be accelerated using implicit imitation learning (IML), a technique that allows the agent to learn a skill through the observation of an expert mentor [10]. For example, IML can be used to teach an agent a surgical skill by observing and imitating the performance of an actual surgeon. The agent observes and analyzes the state transitions of the surgeon’s actions (for example, how the surgeon moves from one position to another, and the composition of every maneuver) and uses the information to update its own states and actions. In a series of works, trajectories recorded from human subjects are used to generate an initial policy (the action generated based on certain state). Additionally, some algorithms for imitation learning can learn from several mentors and are being

used to transfer knowledge between agents with different reward structures [3,10].

Inverse RL (IRL) is a technique for imitation learning that also consists of an observer agent and a mentor. In IRL the agent learns the reward function of the environment from the observer. Then the agent builds the policy that maximizes the reward function using classical RL’s trial and error approach [3].

Another layer of ML is deep learning (DL). DL allows for processing of huge chunks of data to find relationships in sets that are often impossible to explicitly label, such as the pixel in a given set of pictures. The basic architecture of DL consists of what are called neural networks, that are analogous to the neurons and synapses of the human brain, providing much of the ability to learn. Deep learning can be used in combination with the previously mentioned techniques [11].

Other aspects of ML deals with fine tuning the agent to account for the unpredictability of, or to accurately represent physical systems. These techniques include system identification, high-quality rendering [12-14], domain adaptation [15] and iterative learning control [16]. Although these methods have some advantages, they generally require large amounts of data and are still labile to unexpected changes and unaccountable environmental elements [17]. It is important to note that ML is generally used with

other methods in order to optimize robotic capabilities. Even though the control architecture (where ML is mostly needed) governs the action performed based on the current state of the environments, two important components cannot be overlooked; the sensors needed for environment observation and the effectors to perform a certain task. Indeed, surgical interventions include interactions with delicate and deformable structures. So, in order to operate, sensory and motor apparatus are being used to detect, for example, the depth needed for a given maneuver, tissue consistency and to generate the required force and direction of a specific maneuver<sup>[3,4]</sup>. For example, Sozzi et al. used real-time adaptive motion planner (RAMP) to generate collision-free robot motions to avoid obstacles within the workspace<sup>[18]</sup>. For the Smart Tissue Anastomosis Robot (STAR), in order to achieve proper suturing, the team used force sensors and infrared bio-glue to prevent tissue deformation and to guide the robot for needle insertion<sup>[19]</sup>.

Next, we will be dealing with current examples of autonomous surgical robots that have incorporated one or more of the methods presented.

## WHAT ROBOTS DO WE HAVE SO FAR?

Van den Berg et al. employed imitation learning to develop an agent that learns tasks from multiple human demonstrations, optimizing speed and smoothness of task execution. The technique was employed on the Berkeley Surgical Robot and used for knot-tying and drawing figures<sup>[16]</sup>. Schulman et al. also developed an agent that learns by human demonstration using a trajectory transfer method. The agent was able to learn five different types of knots<sup>[20]</sup>. Recently, Calinon et al. used inverse RL to transfer skill from a surgeon teleoperator to a flexible robot. In this method the agent and the mentor may have different morphological structures and still handle the case of skill transfer<sup>[21]</sup>.

Moreover, Mayer et al. apply ML for suturing and knot-tying, using recurrent neural networks (a subset of deep learning), publishing a series of work<sup>[22,23]</sup>. Also, they used imitation learning to create a suturing robot using principles known from fluid dynamics<sup>[24]</sup>.

Weede et al. developed an intelligent autonomous endoscopic guidance system that anticipates the surgeon's next action during a procedure and adjusts the position of the endoscopic camera accordingly. The system uses information on the movements of the instruments from previous procedures<sup>[25]</sup>.

Krieger et al. used The STAR robot to suture bowels in pigs. They used the concept of supervised autonomous suturing, where the surgeon outlines the incision area then the robot uses sensors and combined 3D imaging to assist in suturing of intestinal anastomosis. The STAR robot was

able to place evenly spaced and leak-proof sutures in a trial with live pigs. However, sometimes the surgeons had to make small adjustments to the thread's position for accurate suturing<sup>[19]</sup>. Moreover, Krieger has taught the robot to remove tumors with infrared markers that were used to mark cancerous areas, the robot then excises these parts with, though preliminary, human level accuracy<sup>[5]</sup>.

Mylonas et al. used GMM in designing an algorithm that learns from human demonstration. They created a basic autonomous eFast ultrasound scanning by a robotic manipulator<sup>[26]</sup>. Kassahun et al. used GMM modeled joint probability densities to make their agent capable of understanding the model of interaction between the aorta and the catheter in interventional procedures<sup>[27]</sup>.

Another agent used in interventional procedures developed by Fagogenis et al. used what they called "Haptic vision" to assist in paravalvular leak closure of prosthetic valves. They designed a robotic catheter that can navigate autonomously (using leak locations localized from pre-operative imaging) to the aortic valve and deploy an occluder into the site of a leak. An operator then deploys the occluder. Haptic vision combines machine learning with intracardiac endoscopy and image processing algorithms to form a hybrid imaging and touch sensor. Machine learning was primarily used to enable the catheter to distinguish the blood and tissue from the prosthetic aortic valve<sup>[28]</sup>.

Last but not least, AI agents are being introduced in Urological procedures. The AquaBeam™ robotic system was approved as a water ablative therapy for the resection of the prostate. Although the technology requires human input and image planning, the robotic system assists in resecting, surgeon defined, prostatic tissue using high velocity saline jet while autonomously adjusting various flow rates based on the depth, length and width of the area being resected<sup>[29]</sup>.

## LIMITATIONS AND HOW TO OVERCOME THEM

All current ML technologies used in ARS, variably share some common drawbacks, mainly, the highly unpredictable nature of the physical world, restraints regarding the training environment, and the amount of data and time needed to train an optimal machine. Training and testing an intelligent agent using available models tends to be time-consuming which generally involves manually collecting and labelling huge amounts of data, for example when using supervised learning. This is problematic when the job requires data that are difficult to obtain in large quantities with necessary variability, labels that are difficult to specify, and/or expert knowledge<sup>[30]</sup>. The end goal of overcoming these limitations is, broadly, to create a system that can implement the desired actions

(suturing for example) with great reliability, flexibility and safety<sup>[4]</sup>. The question is whether it is possible to create the optimal environment for ML to overcome the physical constraints of the real world (including the data availability issue), and increase the speed of skill acquisition while being adaptable to random changes and the complex nature of the physical reality. We will be focusing on modeling challenges, data limitation and adaptation to physical changes. The solutions suggested here are based on developments outside the medical field, as we suggest the need to incorporate these techniques in future autonomous surgical robotics designs.

### Modeling challenges and physical data limitations

One of the major challenges in modeling the surgical environment is the deforming and dynamic nature of the living body due to physiological, pathological and even external phenomena. For that purpose, mechanical, geometric, and physiological behavior of the environment should be considered. The current methods that rely on intraoperative inputs are not optimal as they involve theoretical and technical challenges related to the interpretation of sensory information, such as sensor co-registration, synchronization and information fusion, which are highly fragile as well as the need for annotated real world data<sup>[3]</sup>. Moreover, applying ML sometimes employs random exploration, which can be hazardous in any real physical training set. ML might also often require thousands or millions of samples, which could take a tremendous amount of time to collect in a real physical word setting, making it impractical for many applications<sup>[31]</sup>. One way of overcoming this limitation is through learning in simulation.

### Data driven physics simulation

Recent results in learning in simulation are promising for building robots with human-level performance on a number of complex tasks<sup>[32,33]</sup>. Ideally, an agent should learn policies that encode complex behaviors utterly in simulation and apply those policies successfully via physical robots.

Moreover, one can speculate that an optimal simulation should be as faithful as possible to the real world. Here comes the role of physics simulators. High quality physics simulations are being used in computer graphics to replicate the physical world, from dynamic fracture animation to fluids and particles simulations<sup>[34,35]</sup>. Although they do not provide an interactive environment for real time simulation, they could be used as a cornerstone to replicate physical reality. On the other hand, there are other methods that allow for real time interactions within the virtual world. For example, Holden et al. developed a data-driven physics simulation method that supports real time interactions with external objects.

Their method combines ML with subspace simulation techniques which enables a very efficient physics simulation that supports accurate interactions with external objects, surpassing existing models<sup>[36]</sup>. Seunghwan et al. created a physics-based simulation of a human musculoskeletal model composed of a skeletal system and 300 muscles with a control system, creating a reliable simulation of anatomical features with robust control of dynamical systems that generates highly realistic human movements. Also, their model demonstrates how movements are affected in specific pathological conditions such as bone deformities and when applying various prostheses<sup>[37]</sup>. The same authors also formulated a technique called VIPER that creates realistic muscle models that simulate controllable muscle movement and even muscle growth<sup>[38]</sup>. Moreover, one can use MuJoCo physics engine which is commonly used to create advanced virtual environments for ML. MuJoCo was used by OpenAI research to virtually train a robotic hand that can solve a Rubik's cube<sup>[39]</sup>. It is only a matter of time before we reach ultra-realistic real time physics simulators that include complex anatomical and physiological elements. So how do these approaches attempt to replicate reality? The aforementioned models and other approaches sometimes make the simulator to closely match physical reality by performing a variety of techniques including system identification, high-quality rendering<sup>[12-14]</sup>, domain adaptation<sup>[15]</sup> and iterative learning control<sup>[16]</sup>. Although they are the best methods to account for known physical entities and generally do not directly rely on physical data, the problem in these techniques is that they are still suboptimal in accounting for the randomness of the real world, requires large sets of synthetic data impute (both data demanding and time consuming) and many times still requires additional training on real-world data<sup>[17]</sup>.

So, we have techniques to simulate the real world that can be trained fast and does not require real physical data input, now the question is how can we ensure that our model can deal with uncounted randomness, can be trained fast, and does not require large synthetic data input?

### Adaptation to unknown situations and overcoming synthetic data limitation

Any system with decision-making power in the operating room should guarantee the safety of the patient while being able to cope with unpredictable events and the uncertainty of the living body. A critical challenge is to develop intelligent agents that are able to adapt the learned skills to unexpected and novel situations<sup>[3]</sup>. For solving the modelling problem, we suggested the use of virtual physics simulation. Unfortunately, incongruities between reality and simulators make transferring skills

from simulation problematic. For example, system identification, a process used for adjusting the parameters of the simulation to match the characteristics of the real world and the behavior of the physical system (i.e. robot), is error-prone and time-consuming. Even with other techniques such as high-quality rendering<sup>[12–14]</sup>, domain adaptation<sup>[15]</sup> and iterative learning control<sup>[16]</sup>, the real world has physical effects that are hard to model and are not captured by current real-time physics simulators like gear backlash, nonrigidity, fluid dynamics and wear-and-tear. Furthermore, simulators are often unable to produce the noise and richness embedded in the real physical world. These differences, known as the reality gap, are considered bottleneck to the usage of simulated data on real physical robots<sup>[17,39]</sup>.

Bridging the ‘reality gap’ that separates experiments on hardware from simulated robotics might accelerate autonomous robotic developments through improved synthetic data availability. This brings us to domain randomization (DR), a new method for training agents on simulated environments that transfer to the real environment by randomizing rendering in the simulator. In other words, the parameters of the simulation are randomized in a way that with enough variability in the simulator, the real world appears to the agent as just another variation. So, the underlying hypothesis is this: if the variability in simulation is significant enough, agents trained in simulation will generalize to the real world. It is important to note that researchers can also try to use DR in combination with other techniques that optimize the physics simulation, which might improve the results<sup>[17,39]</sup>.

### Domain randomization

In DR, the parameters of the simulator—like lighting, pose, object textures, and other physical aspects—are randomized to oblige the agent to learn the essential features of the object and task of interest. DR requires us to specify what aspects we want to randomize, and specify the variable testing states. Although in its early form, the importance of DR is that it allows for the possibility to produce an agent with strong performance using low-fidelity synthetic data. This introduces the possibility of using inexpensive artificial (synthetic) data for training agents while avoiding the need to collect and label incredible amounts of real-world data or to generate highly realistic artificial worlds<sup>[30]</sup>. One of the earliest works on domain randomization was presented by Tobin et al. The team used DR in the setting of RL and managed to train a robot virtually to localize presented objects. They were able to train an accurate real-world object detector that is resistant to partial occlusions and other distractors using synthetic data from a simulator with non-realistic random textures. The detectors were

also used to perform grasping in a messed up real environment<sup>[17]</sup>. Also, OpenAI used RL with DR to enable a robot to learn dexterous in-hand manipulations<sup>[40]</sup>.

Automatic domain randomization (ADR), is a variation of DR that randomizes the parameters of the simulator automatically, without the need to specify what elements of the simulation we want to change. ADR automatically generates a distribution over randomized environments of ever-increasing difficulty, thus creating millions for scenarios for the learning algorithm. The latest implementation of ADR was conducted by the OpenAI team, who trained a robotic arm to solve a rubik’s cube. The robot was able to solve the cube in a real world setting, even with intentional disturbance of the environment by the researchers<sup>[39]</sup>.

Compared to iterative learning control and domain adaptation which are important tools for addressing the reality gap, DR does not require additional training on real-world data. Although DR requires no additional real world training, it can also be combined easily with most parallel techniques, and we should consider using it in combination with realistic physics simulations and possibly even, when possible, other training methods such as imitation learning<sup>[17]</sup>. DR (or ADR), through randomization, reduces the need of synthetic data while creating an agent robust to changes in the real world. It is important to note that DR is still a new technique and will require further optimization to improve its usefulness and applicability.

In brief, what we are suggesting in this review is to try to combine optimal realistic physical simulation techniques with DR (or ADR) and other parallel techniques in a sense that we can create an environment as faithful to the real world, and as random as the real world in order to produce optimal training environment for surgical agents thus creating the most reliable autonomous surgical robots.

### CONCLUSION

Autonomous robots will be needed to address the decreased work force, the increased demand for surgery and the high risk of medical errors. Current technologies used in medical robotics require huge amounts of data input, are difficult to train, and prone to minor changes in the environment. Using physics simulation techniques combined with DR might be what is needed to overcome these limitations in order to create the advances desirable in autonomous robotic surgery. Our approach focuses on virtual training with domain randomization. Of course, to apply the learned skills in the real world certain specific hardware will be needed, however these technologies are currently available in the form of sensors and other hardware already in use in many current

medical and non-medical models. Here we are adding this layer of training that if linked properly to a physical hardware, the aforementioned limitations should be eventually overcome. In essence, more trial and less error.

## DECLARATIONS

### Conflicts of interest

The authors declare that they have no conflict of interest.

### Authors' contributions:

Dr. Youssef Ghosn, Dr. Mohammed Hussein Kamareddine, Mr. Geroge Salloum, Dr. Elie Najem, Mr. Ayman Ghosn and Dr. Charbel Saade, designed and conceptualized the study and drafted the manuscript for intellectual content. All authors approved the final version and agreed to be accountable for all aspects of the work.

## REFERENCES

- Aruni, G., Amit, G., & Dasgupta, P. (2018). New surgical robots on the horizon and the potential role of artificial intelligence. *Investigative and clinical urology*, 59(4), 221-222.
- Anderson, J. G., & Abrahamson, K. (2017). Your Health Care May Kill You: Medical Errors. In *ITCH* (pp. 13-17).
- Kassahun, Y., Yu, B., Tibebe, A. T., Stoyanov, D., Giannarou, S., Metzner, J. H., & VanderPoorten, E. (2016). Surgical robotics beyond enhanced dexterity instrumentation: a survey of machine learning techniques and their role in intelligent and autonomous surgical actions. *International journal of computer assisted radiology and surgery*, 11(4), 553-568.
- Taylor, R. H., Kazanzides, P., Fischer, G. S., & Simaan, N. (2020). Medical robotics and computer-integrated interventional medicine. In *Biomedical Information Technology* (pp. 617-672). Academic Press.
- Svoboda, E. (2019). Your robot surgeon will see you now. *Nature*, 573, S110-S111.
- Taylor, R. H., Kazanzides, P., Fischer, G. S., & Simaan, N. (2020). Medical robotics and computer-integrated interventional medicine. In *Biomedical Information Technology* (pp. 617-672). Academic Press.
- Kaelbling, L. P., Littman, M. L., & Moore, A. W. (1996). Reinforcement learning: A survey. *Journal of artificial intelligence research*, 4, 237-285.
- Russel, S., & Norvig, P. (2013). *Artificial intelligence: a modern approach*. Pearson Education Limited.
- Hinton, G. E., Sejnowski, T. J., & Poggio, T. A. (Eds.). (1999). *Unsupervised learning: foundations of neural computation*. MIT press.
- Price, B., & Boutillier, C. (2003, August). A Bayesian approach to imitation in reinforcement learning. In *IJCAI* (pp. 712-720).
- Guo, Y., Liu, Y., Oerlemans, A., Lao, S., Wu, S., & Lew, M. S. (2016). Deep learning for visual understanding: A review. *Neurocomputing*, 187, 27-48.
- Planche, B., Wu, Z., Ma, K., Sun, S., Kluckner, S., Lehmann, O., ... & Ernst, J. (2017, October). DepthSynth: Real-time realistic synthetic data generation from cad models for 2.5 d recognition. In *2017 International Conference on 3D Vision (3DV)* (pp. 1-10). IEEE.
- James, S., & Johns, E. (2016). 3d simulation for robot arm control with deep q-learning. *arXiv preprint arXiv:1609.03759*.
- Richter, S. R., Vineet, V., Roth, S., & Koltun, V. (2016, October). Playing for data: Ground truth from computer games. In *European conference on computer vision* (pp. 102-118). Springer, Cham.
- Cutler, M., & How, J. P. (2015, May). Efficient reinforcement learning for robots using informative simulated priors. In *2015 IEEE International Conference on Robotics and Automation (ICRA)* (pp. 2605-2612). IEEE.
- Van Den Berg, J., Miller, S., Duckworth, D., Hu, H., Wan, A., Fu, X. Y., ... & Abbeel, P. (2010, May). Superhuman performance of surgical tasks by robots using iterative learning from human-guided demonstrations. In *2010 IEEE International Conference on Robotics and Automation* (pp. 2074-2081). IEEE.
- Tobin, J., Fong, R., Ray, A., Schneider, J., Zaremba, W., & Abbeel, P. (2017, September). Domain randomization for transferring deep neural networks from simulation to the real world. In *2017 IEEE/RSJ international conference on intelligent robots and systems (IROS)* (pp. 23-30). IEEE.
- Sozzi, A., Bonfè, M., Farsoni, S., De Rossi, G., & Muradore, R. (2019). Dynamic Motion Planning for Autonomous Assistive Surgical Robots. *Electronics*, 8(9), 957.
- Leonard, S., Wu, K. L., Kim, Y., Krieger, A., & Kim, P. C. (2014). Smart tissue anastomosis robot (STAR): A vision-guided robotics system for laparoscopic suturing. *IEEE Transactions on Biomedical Engineering*, 61(4), 1305-1317.
- Schulman, J., Ho, J., Lee, C., & Abbeel, P. (2016). Learning from demonstrations through the use of non-rigid registration. In *Robotics Research* (pp. 339-354). Springer, Cham.
- Calinon, S., Bruno, D., Malekzadeh, M. S., Nanayakkara, T., & Caldwell, D. G. (2014). Human-robot skills transfer interfaces for a flexible surgical robot. *Computer methods and programs in biomedicine*, 116(2), 81-96.
- Mayer, H., Gomez, F., Wierstra, D., Nagy, I., Knoll, A., & Schmidhuber, J. (2008). A system for robotic heart surgery that learns to tie knots using recurrent neural networks. *Advanced Robotics*, 22(13-14), 1521-1537.
- Mayer, H., Nagy, I., Burschka, D., Knoll, A., Braun, E. U., Lange, R., & Bauernschmitt, R. (2008, March). Automation of manual tasks for minimally invasive surgery. In *Fourth International Conference on Autonomic and Autonomous Systems (ICAS'08)* (pp. 260-265). IEEE.
- Mayer, H., Nagy, I., Knoll, A., Braun, E. U., Lange, R., & Bauernschmitt, R. (2007, April). Adaptive control for human-robot skill transfer: Trajectory planning based on fluid dynamics. In *Proceedings 2007 IEEE International Conference on Robotics and Automation* (pp. 1800-1807). IEEE.

25. Weede, O., Mönnich, H., Müller, B., & Wörn, H. (2011, May). An intelligent and autonomous endoscopic guidance system for minimally invasive surgery. In 2011 IEEE International Conference on Robotics and Automation (pp. 5762-5768). IEEE.
26. Mylonas, G. P., Giataganas, P., Chaudery, M., Vitiello, V., Darzi, A., & Yang, G. Z. (2013, November). Autonomous eFAST ultrasound scanning by a robotic manipulator using learning from demonstrations. In 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems (pp. 3251-3256). IEEE.
27. Kassahun, Y., Yu, B., & Vander Poorten, E. (2013). Learning catheter-aorta interaction model using joint probability densities. In Proceedings of the 3rd joint workshop on new technologies for computer/robot assisted surgery (pp. 158-160).
28. Fagogenis, G., Mencattelli, M., Machaidze, Z., Rosa, B., Price, K., Wu, F., ... & Dupont, P. E. (2019). Autonomous robotic intracardiac catheter navigation using haptic vision. *Science robotics*, 4(29), eaaw1977.
29. Navaratnam, A., Abdul-Muhsin, H., & Humphreys, M. (2018). Updates in urologic robot assisted surgery. *F1000Research*, 7.
30. Tremblay, J., Prakash, A., Acuna, D., Brophy, M., Jampani, V., Anil, C., ... & Birchfield, S. (2018). Training deep networks with synthetic data: Bridging the reality gap by domain randomization. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (pp. 969-977).
31. Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., ... & Petersen, S. (2015). Human-level control through deep reinforcement learning. *Nature*, 518(7540), 529-533.
32. Levine, S., Finn, C., Darrell, T., & Abbeel, P. (2016). End-to-end training of deep visuomotor policies. *The Journal of Machine Learning Research*, 17(1), 1334-1373.
33. Schulman, J., Levine, S., Abbeel, P., Jordan, M., & Moritz, P. (2015, June). Trust region policy optimization. In International conference on machine learning (pp. 1889-1897).
34. Wolper, J., Fang, Y., Li, M., Lu, J., Gao, M., & Jiang, C. (2019). CD-MPM: continuum damage material point methods for dynamic fracture animation. *ACM Transactions on Graphics (TOG)*, 38(4), 1-15.
35. Gao, M., Pradhana, A., Han, X., Guo, Q., Kot, G., Sifakis, E., & Jiang, C. (2018). Animating fluid sediment mixture in particle-laden flows. *ACM Transactions on Graphics (TOG)*, 37(4), 1-11.
36. Holden, D., Duong, B. C., Datta, S., & Nowrouzezahrai, D. (2019, July). Subspace neural physics: fast data-driven interactive simulation. In Proceedings of the 18th annual ACM SIGGRAPH/Eurographics Symposium on Computer Animation (pp. 1-12).
37. Lee, S., Park, M., Lee, K., & Lee, J. (2019). Scalable muscle-actuated human simulation and control. *ACM Transactions on Graphics (TOG)*, 38(4), 1-13.
38. Angles, B., Rebain, D., Macklin, M., Wyvill, B., Barthe, L., Lewis, J., ... & Tagliasacchi, A. (2019). Viper: Volume invariant position-based elastic rods. *Proceedings of the ACM on Computer Graphics and Interactive Techniques*, 2(2), 1-26.
39. Li, T., Xi, W., Fang, M., Xu, J., & Meng, M. Q. H. (2019). Learning to Solve a Rubik's Cube with a Dexterous Hand. arXiv preprint arXiv:1907.11388.
40. Andrychowicz, O. M., Baker, B., Chociej, M., Jozefowicz, R., McGrew, B., Pachocki, J., ... & Schneider, J. (2020). Learning dexterous in-hand manipulation. *The International Journal of Robotics Research*, 39(1), 3-20.