Detection, Segmentation, and 3D Pose Estimation of Surgical Tools Using Convolutional Neural Networks and Algebraic Geometry

Md. Kamrul Hasan[a,b], Lilian Calvet[a], Navid Rabbani[a], Adrien Bartoli[a]

[a]*EnCoV, Institut Pascal, UMR 6602 CNRS/Université Clermont-Auvergne, Clermont-Ferrand, France*
[b]*Department of Electrical and Electronic Engineering, Khulna University of Engineering & Technology, Khulna-9203, Bangladesh*

Abstract

Background and objective

Surgical tool detection, segmentation, and 3D pose estimation are crucial components in Computer-Assisted Laparoscopy (CAL). The existing frameworks have two main limitations. First, they do not integrate all three components. Integration is critical; for instance, one should not attempt computing pose if detection is negative. Second, they have highly specific requirements, such as the availability of a CAD model. We propose an integrated and generic framework whose sole requirement for the 3D pose is that the tool shaft is cylindrical. Our framework makes the most of deep learning and geometric 3D vision by combining a proposed Convolutional Neural Network (CNN) with algebraic geometry. We show two applications of our framework in CAL: tool-aware rendering in Augmented Reality (AR) and tool-based 3D measurement.

Methods

We name our CNN as ART-Net (Augmented Reality Tool Network). It has a Single Input Multiple Output (SIMO) architecture with one encoder and multiple decoders to achieve detection, segmentation, and geometric primitive extraction. These primitives are the tool edge-lines, mid-line, and tip. They allow the tool's 3D pose to be estimated by a fast algebraic procedure. The framework only proceeds if a tool is detected. The accuracy of segmentation and geometric primitive extraction is boosted by a new Full resolution feature map Generator (FrG). We extensively evaluate the proposed framework with the EndoVis and new proposed datasets. We compare the segmentation results against several variants of the Fully Convolutional Network (FCN) and U-Net. Several ablation studies are provided for detection, segmentation, and geometric primitive extraction. The proposed datasets are surgery videos of different patients.

Results

In detection, ART-Net achieves 100.0 % in both average precision and accuracy. In segmentation, it achieves 81.0 % in mean Intersection over Union (mIoU) on the robotic EndoVis dataset (articulated tool), where it outperforms both FCN and U-Net, by 4.5 *pp* and 2.9 *pp*, respectively. It achieves 88.2 % in mIoU on the remaining datasets (non-articulated tool). In geometric primitive extraction, ART-Net achieves 2.45° and 2.23° in mean Arc Length (mAL) error for the edge-lines and mid-line, respectively, and 9.3 pixels in mean Euclidean distance error for the tool-tip. Finally, in terms of 3D pose evaluated on animal data, our framework achieves 1.87mm, 0.70mm, and 4.80mm mean absolute errors on the X, Y, and Z coordinates, respectively, and 5.94° angular error on the shaft orientation. It achieves 2.59mm and 1.99mm in mean and median location error of the tool head evaluated on patient data.

Conclusions

The proposed framework outperforms existing ones in detection and segmentation. Compared to separate networks, integrating the tasks in a single network preserves accuracy in detection and segmentation but substantially improves accuracy in geometric primitive extraction. Overall, our framework has similar or better accuracy in 3D pose estimation while largely improving robustness against the very challenging imaging conditions of laparoscopy. The source code of our framework and our annotated dataset will be made publicly available at `https://github.com/kamruleee51/ART-Net`.

Keywords. Computer-Assisted Laparoscopy, Augmented Reality, Deep Learning, Segmentation, 3D Pose, Algebraic Geometry.

1. Introduction

Laparoscopy is a preferable approach for many surgical procedures, as it reduces blood loss, trauma, infection rate, and increases the speed of recovery compared to open surgery (Buell et al., 2008; Cheung et al., 2013; Fuks et al., 2016; Jaffray, 2005). However, it imposes technical difficulties to the surgeon, including hand-eye disalignment and a narrow field of view. Automatic surgical tool detection, segmentation, and 3D pose estimation can be of great benefit to CAL. However, these tasks are very challenging due to the presence of smoke, blood, partial occlusions, shadows, specularities, motion blur, gauze, and complex background textures (Attia et al., 2017; Garcia-Peraza-Herrera et al., 2016; Pakhomov et al., 2019), as illustrated in figure 1.

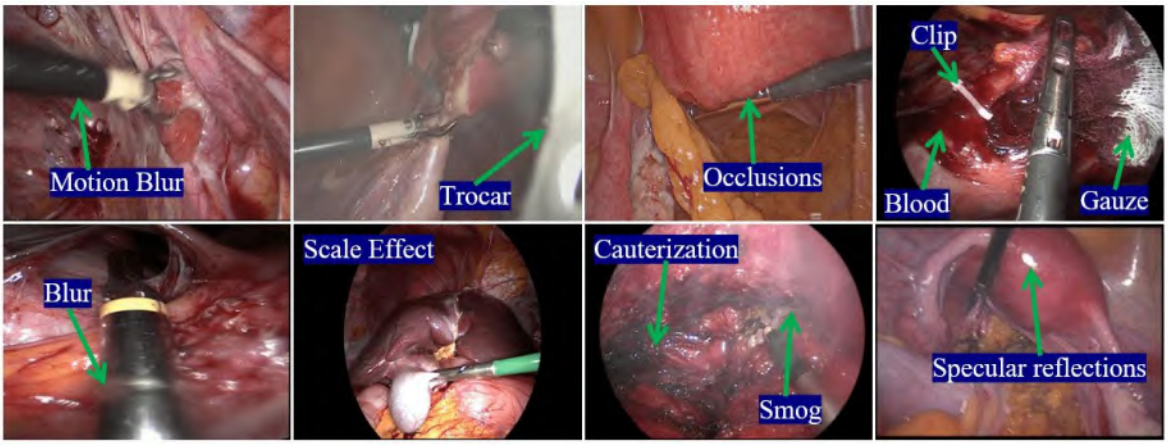


Figure 1: Examples of challenging laparoscopic conditions for automatic tool detection, segmentation, and 3D pose estimation. The results from the proposed ART-Net on these images are shown in Appendix A.

A large number of surgical tool segmentation methods have already been proposed. Methods based on discriminant color features or structural feature descriptors were proposed by Agustinos and Voros (2015); Allan et al. (2012); Doignon et al. (2005). These techniques were recently outperformed by deep learning methods (Garcia-Peraza-Herrera et al., 2017, 2016; Pakhomov et al., 2019), which nonetheless still have limitations, including the loss of image details, lack of robustness to perturbations, and inability to handle tools generically. In pose estimation, it is essential to distinguish between 2D and 3D pose. The former refers to the 2D position of the tool body and joints in the 2D image, whereas the latter refers to the

3D position and orientation of the tool in the camera's 3D coordinate frame. Our method's scope is 3D pose estimation, for which using laparoscopic images proved more accurate than tracking devices (Feuerstein et al., 2007). Feature-based methods were proposed by Salah et al. (2011); Wang et al. (2013). These methods are computationally expensive and inaccurate. Jayarathne et al. (2013); Pratt et al. (2015) use fiducial markers placed on an intraoperative ultrasound probe and retrieve 3D pose by solving the Perspective-n-Point (PnP) problem. However, the currently available tools do not have printed markers on them, which make these methods unusable in routine surgery. A useful, practical system must hence be able to handle markerless tools. The image-based generic tool detection, segmentation, and 3D pose estimation thus still form essential open problems.

We propose a new integrated framework combining a CNN with algebraic geometry for generic tool detection, segmentation, and 3D pose estimation. It is depicted in figure 2. Our framework uses our CNN ART-Net to solve detection, segmentation, and geometric primitive extraction efficiently. It then uses the geometric primitives to compute 3D pose with algebraic geometry. This last step is efficient because it uses a simple physics-based model, for which there is no reason to use a learned approximation. ART-Net detects the tool, segments the image, and extracts the geometric primitives with a single encoder. It is trained in an end-to-end fashion. The proposed framework has other advantages: it avoids outputting in 3D pose space from the CNN[1] and does not require 3D pose ground-truth, which would be extremely difficult to obtain accurately in practice. We validate our framework on *in-vivo* laparoscopic images, showing that it outperforms existing approaches.

The paper is organized as follows. Section 2 presents the state-of-the-art. Section 3 presents the methodology and materials. Section 4 reports the experiments and results. Section 5 concludes.

[1] 3D pose lives on a nonlinear manifold with complex topology.

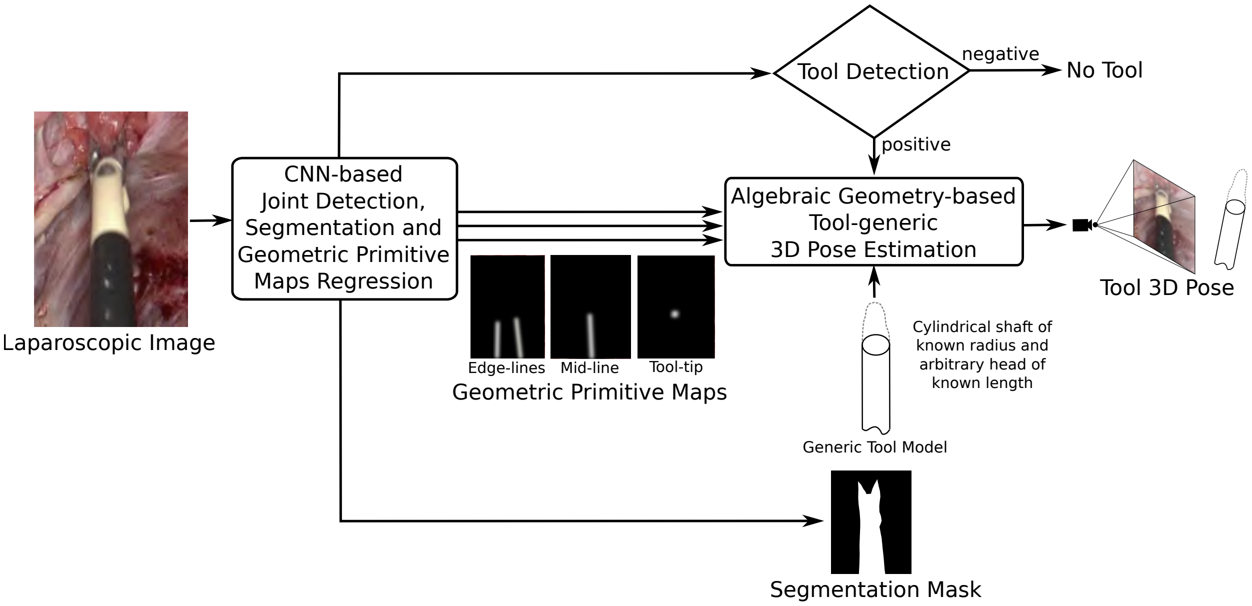

Laparoscopic Image
CNN-based Joint Detection, Segmentation and Geometric Primitive Maps Regression
Tool Detection
negative
No Tool
positive
Algebraic Geometry-based Tool-generic 3D Pose Estimation
Tool 3D Pose
Edge-lines
Mid-line
Tool-tip
Geometric Primitive Maps
Cylindrical shaft of known radius and arbitrary head of known length
Generic Tool Model
Segmentation Mask

Figure 2: The proposed framework for concurrent tool detection, segmentation, and geometric primitive extraction for 3D pose estimation.

2. Previous Works and Contributions

We survey previous work on surgical tool detection, segmentation, and 3D pose estimation in the following three sections; and finally, our contributions.

2.1. Surgical Tool Detection

The early detection methods used color-marker extraction via low-level image processing to detect the shaft or the tip of the tool (Krupa et al., 2003; Wei et al., 1997). These methods are accurate in tracking and efficient in computation but fail to deal with strong color and lighting variations. Some other techniques exploit gradient-based primitives and geometric constraints to identify the tool shaft (Agustinos and Voros, 2015). However, tool edges very often blend with the background, defeating these methods. Since then, CNN has outperformed other methods in object detection (Arel et al., 2010). An EndoNet architecture was proposed by Twinanda et al. (2016) for both tool presence detection and phase recognition. It is an extension of the AlexNet architecture (Krizhevsky et al., 2012) which contains eight layers, namely five convolutional layers followed by three fully-connected layers. The confidence probabilities of EndoNet are associated with the tool categories used for the tool presence detection. Choi et al. (2017) applied the YOLO architecture along with transfer learning from ImageNet (Deng et al., 2009) but reported precision of only 72.26 %. A deep learning-based multi-label classification method for surgical tool presence detection was proposed by Wang et al. (2017), which combines the VGG (Simonyan and Zisserman, 2014) and GoogleNet (Szegedy et al., 2015) networks. Kurmann et al. (2017) proposed a CNN-based method performing tool detection and 2D pose estimation jointly. Jin et al. (2018) leveraged region-based CNNs (R-CNNs), but processing speed was limited to 5 fps, making it unusable for real-time applications. Al Hajj et al. (2018) monitored the tool presence automatically during surgery using CNN and Recurrent Neural Networks (RNNs), where the CNN outputs feed the RNN to create temporal relationships between events. The training is, however, not end-to-end as, due to computational complexity, CNN and RNN are trained independently. Nwoye et al. (2019) developed an end-to-end approach comprising a CNN and a Convolutional LSTM (ConvLSTM) network. They apply the ConvLSTM to model the temporal dependencies in the motion of the surgical tools. Jin et al. (2020) proposed a Multi-task Recurrent Convolutional Network with Correlation Loss (MTRCNet-CL) for tool presence detection and surgical phase recognition. The combined use of low and high-level features leads to improved results for both the detection and recognition tasks. Although deep learning has overall greatly improved surgical tool detection over the recent decade, there is room for improvement. In particular, none of the existing detectors has been integrated into the architecture, including segmentation and geometric primitive extraction end-to-end.

2.2. Surgical Tool Segmentation

Allan et al. (2012) proposed a probabilistic supervised segmentation method using Random Forest (RF) trained on hue and saturation without post-processing. Agustinos and Voros (2015) used color and shape information of the surgical tool based on CIELab and *Cab*. This is followed by automatic Otsu thresholding, skeletonization, and morphological erosion. Finally, a contour detection algorithm (Suzuki et al., 1985) was used to extract the tool contour for each region as an oriented bounding box.

More recently, the use of deep learning has substantially boosted segmentation performance, especially with the introduction of the U-Net (Ronneberger et al., 2015), as reported by Garcia-Peraza-Herrera et al. (2017, 2016); Laina et al. (2017); Pakhomov et al. (2019). An automatic real-time method based on an FCN with an improved learning process was proposed by Garcia-Peraza-Herrera et al. (2016). The Cyclical Learning Rate (CLR) (Smith, 2017) and the pre-trained model on the PASCAL-context dataset (Mottaghi et al., 2014) were used to improve segmentation accuracy. Attia et al. (2017) applied a hybrid-CNN method utilizing both recurrent and convolutional networks simultaneously. To prevent the coarse segmentation (Pakhomov et al., 2019), a Recurrent Neural Network (RNN) was trained to model contextual relationships between pixels, where four layers of RNN were used to find local and global dependencies between pixels in coupled directions. Garcia-Peraza-Herrera et al. (2017) proposed two novel deep learning architectures for the automatic segmentation of non-rigid surgical tools, namely ToolNetMS and ToolNetH. In ToolNetMS, all scales in FCN8s were summed in a cascaded fashion to ensure better responses around the edges than traditional FCN8s. In contrast, ToolNetH aggregates all the cross-entropy losses. Pakhomov et al. (2019) employed deep residual learning and dilated convolutions, where the coarse segmentation was mitigated by setting strides equal to one in the last two convolution layers. However, as in tool detection, tool segmentation remains an open problem when considering the wide variety of image disturbances occurring in laparoscopy and the complex behavior of light reflection on the tool material.

2.3. Surgical Tool 3D Pose Estimation

Jayarathne et al. (2013); Pratt et al. (2015) compute the 3D pose of an intraoperative ultrasound probe. In (Jayarathne et al., 2013), an 'X-corner' fiducial marker is attached to the probe head, providing up to 11 3D-2D correspondences. In (Pratt et al., 2015), a planar marker made up of black circular dots is used, providing a set of 21 3D-2D correspondences. For both methods, the 3D pose is then obtained using PnP. One of their main advantages is to provide an unambiguous pose compared to the pose obtained from the image of a uniform cylindrical shaft, for which there is an obvious rotational ambiguity. Their main drawbacks are twofold: the requirement for modifying the tool to add the marker and a strong limitation put on the range of viewpoints from which the marker is well visible.

Allan et al. (2012) and Allan et al. (2015) use RF to segment the tool. The segmented tool region then initializes an energy minimization algorithm for estimating the pose of a prior 3D model of the tool within a level set framework. The errors were evaluated *ex-vivo* with a lamb liver as background. In (Allan et al., 2012), the method is tool-generic, simply assuming a cylindrical tool shaft. The obtained errors are up to about $15mm$ in X and Y and up to about $50mm$ in Z. In (Allan et al., 2015), the method uses a CAD model of a robotic tool. For some images, to deal with fast motion, the imaged tool is tracked frame-to-frame using the Lucas-Kanade algorithm. In that case, the 3D pose estimation is formulated as 3D-2D registration, using pixel motions from a reference image for which the tool 3D pose was computed. The mean errors on tip points are about $1mm$ in X and Y and $7mm$ in Z. Agustinos and Voros (2015) use morphological operations and a distance transform to compute tool bounding boxes. Frangi filters are then used to robustly extract tool edges, which are subsequently used as geometric primitives to estimate 3D pose, assuming a cylindrical tool shaft of the known radius. The 3D errors are evaluated *ex-vivo* using a commercial robotic tool holder and a printout of a surgical scene as a background. The mean errors are about $2mm$ in X and Y and $7mm$ in Z.

Accurately estimating the tool's 3D pose remains an open and challenging problem. Existing tool-generic methods report prohibitive depth errors for a large number of CAL applications, such as registration guidance in monocular augmented laparoscopy and specification of 3D points to measure anatomical structures. Besides, the errors were reported from *ex-vivo* evaluations, and their translation to real conditions in terms of accuracy and robustness is unpredictable.

2.4. Contributions

We propose the first integrated framework addressing all the above-mentioned shortcomings of the existing methods. Our framework combines the proposed ART-Net with algebraic geometry. We propose technical solutions and innovations to address each of its steps. We provide a comprehensive quantitative validation and comparison to existing methods, including, for the first time, a quantitative evaluation of 3D pose on real surgery data. We finally show how our framework contributes to two concrete CAL applications.

Technically, ART-Net has a single encoder and five sub-network branches, namely one for tool detection, one for tool segmentation, and three for geometric primitive extraction. The encoder is a convolution network with five blocks and thirteen layers following VGG-16 (Simonyan and Zisserman, 2014). To make inference robust and lightweight for real-time AR applications, we use depth-wise separable convolution in the sub-networks, inspired by the Xception network (Chollet, 2017) and Kaiser et al. (2017). To retrieve the lost spatial and edge information due to subsampling, reduce checkerboard noise (Odena et al., 2016), and minimize over-segmentation, we introduce a Feature map Generator (FrG), a novel special skip connection. FrG can be optionally added to skip connections in a ladder-like structure inspired by the U-Net. It connects the first layer of the encoder to the decoder's last layer and is composed of several depth-wise separable convolutions. As such, it does not include subsampling. The numbers of foreground and background pixels broadly differ, which creates an imbalance in the segmentation task and is known to restrict cross-entropy performance. We address this issue by combining cross-entropy with the IoU in a single loss function, where, specifically, the IoU strengthens the otherwise neglected foreground pixels.

The geometric primitives extracted by ART-Net are then used for 3D pose estimation using algebraic geometry. We use advanced projective geometry to solve the inverse problem of 3D pose elegantly and, importantly, without approximating the physics-based pin-hole camera model. This results in a fast and straightforward algebraic procedure.

3. Proposed Framework

We present the proposed design of ART-Net and subsequent 3D pose estimation, the detailed ART-Net architecture, and the detailed 3D pose procedure.

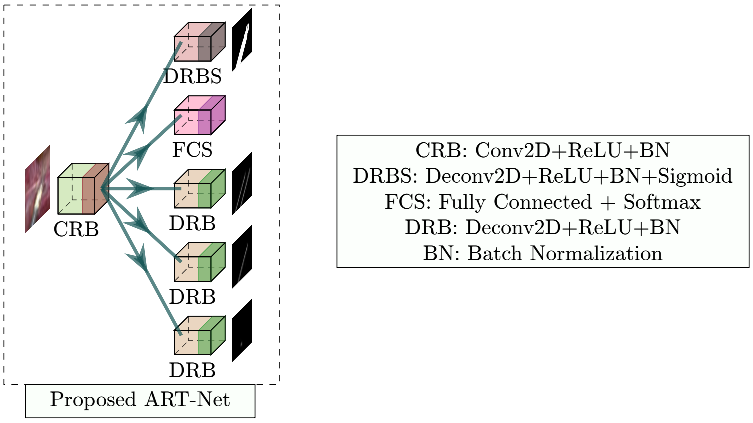
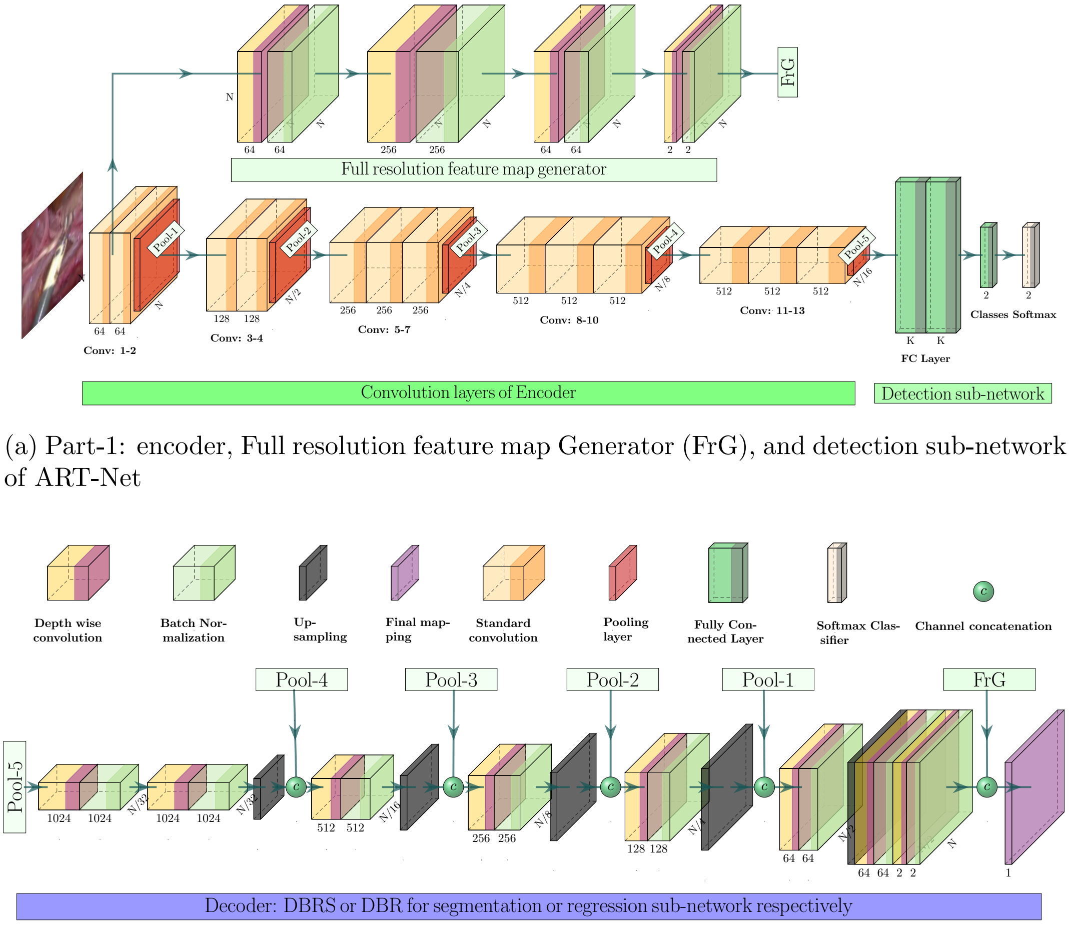


Figure 3: Overview of the proposed ART-Net. The primitives from three DRB regression blocks and the FCS block tool flag are utilized for the 3D pose estimation.

3.1. Pipeline

A global schematic of ART-Net is shown in figure 3. The encoder is composed of convolutions, *relu* activation functions, subsampling, and Batch Normalisation (BN). It is referred to as CRB, standing for Conv+ReLU+BN. It feeds a fully connected layer connected to a *softmax* classifier outputting tool detection, referred to as FCS, standing for Fully Connected+Softmax, and four decoders, one for tool segmentation and three for geometric primitives extraction. The decoders are composed of deconvolutions, *relu* activation functions, up-sampling, and BN. The decoder dedicated to segmentation outputs the probability of pixels belonging to the imaged tool. These probabilities are computed using a sigmoid function. This decoder is referred to as DRBS, standing for Deconv+ReLU+BN+Sigmoid. The three decoders dedicated to geometric primitive extraction are regression sub-networks outputting the geometric primitive maps $\mathcal{I}_*$ with $* \in \{el, ml, tt\}$, which are the truncated distance transforms of the edge-lines, mid-line, and tool-tip, respectively. The design of all the sub-networks is described below in detail.

If the tool is detected, the geometric primitives required to compute the tool 3D pose are extracted from the maps $\mathcal{I}_*$. An initial estimate of the pose is then calculated from the tool shaft's known radius using algebraic geometry. The obtained estimate is finally refined by minimizing the reprojection errors, obtained by directly interpolating the pixel values in $\mathcal{I}_*$.



(a) Part-1: encoder, Full resolution feature map Generator (FrG), and detection sub-network of ART-Net

(b) Part-2: decoder for segmentation or regression sub-networks of ART-Net

Figure 4: The two different parts of the proposed ART-Net. The complete architecture replicates Part-2 four times to create the segmentation and the three geometric primitive extraction sub-networks.

3.2. Detection, Segmentation, and Geometric Primitive Extraction Sub-networks

The detailed ART-Net architecture is shown in figure 4. For the sake of clarity, it has been divided into two parts. The complete assembled architecture is available in GitHub (Hasan et al., 2020).

General principle. In general, a CNN for semantic segmentation has an encoder and a decoder with pixel-wise classification (Badrinarayanan et al., 2017; Long et al., 2015; Ronneberger et al., 2015). The encoder is composed of convolution and subsampling layers and performs feature extraction (Lin et al., 2013). The subsampling layers achieve spatial invariance by reducing the feature maps. This reduction leads to the extraction of more features and reduces the computation cost (Long et al., 2015). The decoder projects the features onto the pixel space to obtain a dense pixel-wise classification (Garcia-Garcia et al., 2018). The reduced feature map typically suffers from the loss of spatial resolution, introducing coarseness, restricted edge information, checkerboard artifacts, and over-segmentation in the segmentation (Long et al., 2015; Odena et al., 2016; Ronneberger et al., 2015).

Resolution problems. To overcome the resolution problems, we propose a special skip connection called FrG, that connects the last layer of the decoder with the original image via a stack of depth-wise separable convolutions without subsampling, as shown in figure 4. FrG provides a full resolution feature map, which compensates for the lost spatial information in the segmentation and regression sub-networks. Additionally, we use traditional skip connections between the corresponding feature maps in the encoder and decoders, with a ladder-like structure (Rasmus et al., 2015) inspired by U-Net.

Preventing overfitting. In existing CNN-based image classifiers, a *flatten* layer is used to vectorize the 2D arrays into a single long continuous linear vector and is followed by several densely connected layers. Most of the parameters of such classifiers belong to the fully connected layers and can cause overfitting. To overcome this limitation, a *dropout* layer (Srivastava et al., 2014) is used as a regulariser, which randomly sets half of the activation of the fully connected layers to zero during training. It improves generalization and prevents overfitting. Lin et al. (2013) proposed a Global Average Pooling (GAP) layer, where only one feature map is generated for each corresponding category. GAP layers perform a more extreme type of dimensionality reduction to avoid overfitting (Lin et al., 2013). In GAP, an $height \times weight \times depth$ dimensional tensor is reduced to a $1 \times 1 \times depth$ vector, similarly to Global Max Pooling (GMP), where each feature map of size $height \times width$ transfers to a single scalar value, namely the average of all the $height * width$ values. As shown in (Lin et al., 2013), GAP is however more robust to spatial translations of the input than GMP. This is because GAP explicitly enforces the feature maps to be confidence maps of categories, as all the spatial regions contribute to the output, while GMP only considers the maximum value over the regions. In our detection sub-network, we used GAP instead of the traditional *flatten* layer, which improves performance in existing image classification methods. We also used *dropout* followed by a *softmax* classifier to detect the surgical tool presence. Besides, the use of GAP contributes to the lightweight design of ART-Net.

Limiting network size. The lightweight ART-Net is achieved by using depth-wise separable convolution (Chollet, 2017) instead of traditional standard convolution. Depth-wise separable convolution is a spatial convolution performed independently over each channel of the input and followed by a point-wise convolution, a 1×1 convolution, which projects the output of the channel by the depth-wise convolution onto a new channel space. For any convolution layer, if we have F filters, M depths, and D_K as kernel size, the total numbers of parameters will be $F * M * D_K^2$ and $M * (F + D_K^2)$ for standard and depth-wise separable convolution, respectively. Thus, the number of parameters in ART-Net is reduced by a factor of $(1/F + 1/D_K^2)$ compared to what it would be with standard convolution.

3.3. 3D Pose Estimation

We first describe our geometric model and notation, then our algebraic procedure to initialize pose, and finally our optimal pose refinement method.

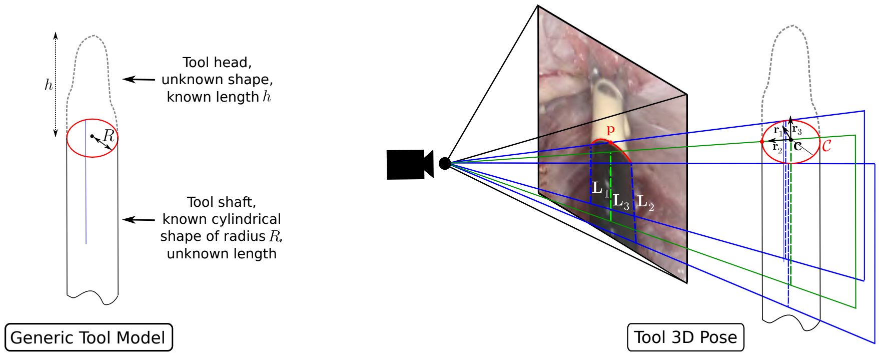


Figure 5: Geometric model used in tool 3D pose estimation. We use perspective projection. The tool model is a cylindrical shaft with a known radius R and a head size h. The edge lines $\mathbf{L}_1$ and $\mathbf{L}_2$ are the occluding contours of the shaft and the mid-line $\mathbf{L}_3$ is the projection of the shaft axis. The shaft orientation is defined by the rotation matrix $\mathsf{R} = [\mathbf{r}_1\ \mathbf{r}_2\ \mathbf{r}_3]$, $\mathbf{r}_3$ being colinear to the shaft axis and $\mathbf{r}_2$ pointing towards the optical center.

3.3.1. Model and Notation

The geometric model used in tool 3D pose estimation is displayed in figure 5. Our generic tool model is composed of two elements: the shaft and the head. The shaft is a cylinder of a known radius of R but an unknown length, while the head has an arbitrary unknown shape and a known length of h. This simple model applies to almost any surgical tool used in laparoscopy and yet allows us to obtain 3D pose from simple geometric primitives. The sought tool pose is represented by $(\mathsf{R}, \mathbf{c}) \in SO(3) \times \mathbb{R}^3$. The center $\mathbf{c}$ of the circle defining the shaft extremity adjoining the tool head defines the tool model origin in camera coordinates, as shown on the right in figure 5. The tool orientation is represented by the rotation matrix R whose three columns $\mathbf{r}_1$, $\mathbf{r}_2$, and $\mathbf{r}_3$ represent the base vectors of the tool in camera coordinates. We choose the third column $\mathbf{r}_3$ as the shaft axis. For any vector-pair so that $[\mathbf{r}'_1\ \mathbf{r}'_2\ \mathbf{r}_3] \in SO(3)$, the tool orientation has then one free rotational degree of freedom around the shaft axis, generated by an angle α as $\mathsf{R}(\alpha) = [\cos(\alpha)\mathbf{r}'_1 - \sin(\alpha)\mathbf{r}'_2\ \ \sin(\alpha)\mathbf{r}'_1 + \cos(\alpha)\mathbf{r}'_2\ \ \mathbf{r}_3]$. We will choose the free angle α for convenience so that $\mathbf{r}_2 = \sin(\alpha)\mathbf{r}'_1 + \cos(\alpha)\mathbf{r}'_2$ is on the plane formed by the shaft axis and the camera center, and directed to the camera center.

We use the pinhole camera model, which is known to work well in laparoscopy (Melo et al., 2011). This model is a mere perspective projection with matrix $\mathsf{K}[\mathbf{I}_3\ \mathbf{0}_3]$ in homogeneous coordinates, where $\mathsf{K} \in \mathbb{R}^{3\times 3}$ represents the camera intrinsics. We calibrate the intrinsic camera parameters, namely the effective focal lengths f_u, f_v and principal point u_0, v_0, and optical distortion using the software Agisoft Lens by filming a checkerboard at the start of surgery. The laparoscopic images are then undistorted to remove the effect of radial and tangential distortion before geometric primitive extraction, and the geometric coordinates are normalized using K^{-1} to 'undo' the effect of K, leaving $[\mathbf{I}_3\ \mathbf{0}_3]$ as projection matrix.

3.3.2. Initialisation of 3D Pose

The initialization of 3D pose is shown in figure 6. It represents a suboptimal estimate necessary to start the optimal refinement given in the next section. Concretely, our initialization procedure follows direct and straightforward steps involving convex optimization. It has two main stages, the computation of the geometric primitive coordinate vectors and the

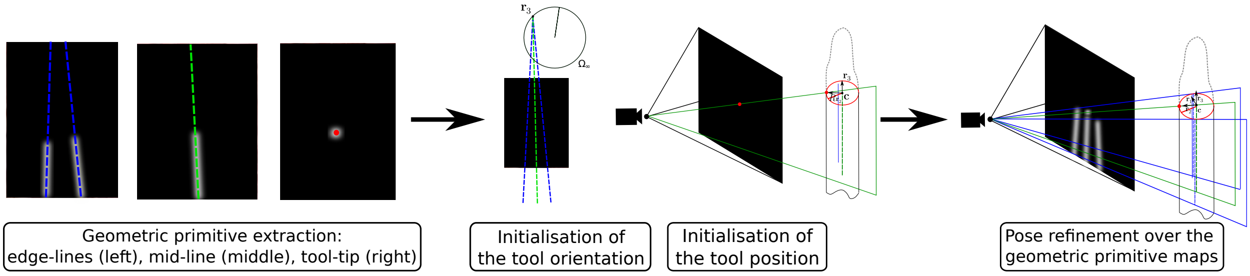

Geometric primitive extraction:
edge-lines (left), mid-line (middle), tool-tip (right)
Initialisation of the tool orientation
Initialisation of the tool position
Pose refinement over the geometric primitive maps
r3
Ω∞
c

Figure 6: Pose estimation pipeline. From left to right: (i) the edge-lines and mid-line are extracted by binarising the geometric primitive maps through thresholding and applying hough line detection; the tool-tip point is extracted as the pixel of maximum value ; (ii) the direction $\mathbf{r}_3$ of the shaft axis is computed as the intersection of the edge-lines and mid-line, while the tool origin $\mathbf{c}$ is computed so that the 3D tool-tip reprojects on the extracted one; (iii) the 3D pose is refined from the geometric primitive maps.

computation of 3D pose from these. The latter stage involves, more specifically, rounds of Linear Least-Squares optimization, solved with standard linear algebra, and rounds of Non-Linear Least-Squares optimization for a low number of unknowns, solved with proposed *ad hoc* closed-forms.

Computation of the geometric primitive coordinate vectors. The first stage is to compute the coordinate vectors representing the geometric primitives in the image. Specifically, the two edge-lines and mid-line are represented by their homogeneous coordinates $\mathbf{L}_1, \mathbf{L}_2, \mathbf{L}_3 \in \mathbb{R}^3$ and the tool-tip is represented by $\mathbf{p} \in \mathbb{R}^2$. The lines $\mathbf{L}_1, \mathbf{L}_2$ and $\mathbf{L}_3$ are extracted by binarising the geometric primitive maps $\mathcal{I}_{el}$ and $\mathcal{I}_{ml}$ through thresholding and applying hough line detection (Ballard, 1981). The point $\mathbf{p}$ is extracted from the geometric primitive map $\mathcal{I}_{tt}$ as the pixel of maximum value.

Computation of 3D pose. The pose, represented by R and $\mathbf{c}$, is then estimated from $\mathbf{L}_1, \mathbf{L}_2$, $\mathbf{L}_3$ and $\mathbf{p}$ through two rounds of linear least-squares optimisations over $\mathbf{r}_3$ and $\mathbf{c}$, respectively. These correspond to the first step of orientation initialization and the second step of position initialization.

We first initialize orientation, represented by $\mathbf{r}_3$, as the vanishing point intersection of $\mathbf{L}_1$, $\mathbf{L}_2$ and $\mathbf{L}_3$. We consider that $\mathbf{L}_1$, $\mathbf{L}_2$, and $\mathbf{L}_3$ are first normalized so that $\mathbf{L}_i^\top \mathbf{X}$ is equal to the Euclidean distance between a point $\mathbf{X}$ in homogeneous coordinates and the line $\mathbf{L}_i$.

The initial estimate $\mathbf{r}_3$ is then obtained from:

$$\mathbf{r}_3 = \arg \min_{\substack{\mathbf{X}\in\mathbb{R}^3 \\ \|\mathbf{X}\|=1}} \|\mathsf{A}\mathbf{X}\|^2 \quad \text{where} \quad \mathsf{A} = \begin{bmatrix} \mathbf{L}_1^\top \\ \mathbf{L}_2^\top \\ \mathbf{L}_3^\top \end{bmatrix} \in \mathbb{R}^{3\times 3}.$$

An elegant solution to this problem is given by taking the singular vector associated with the smallest singular value of A, from its Singular Value Decomposition (SVD). We finally compute $\mathbf{r}_1$ and $\mathbf{r}_2$ as follows. Recall that $\mathbf{r}_1$ is normal to the plane through the optical center and the shaft axis. The equation of this plane, noted π_1, is $[\mathbf{L}_1^\top + \mathbf{L}_2^\top, 0]^\top$, representing the backprojection of the equidistant line to the lines $\mathbf{L}_1$ and $\mathbf{L}_2$ through camera projection $[\mathbf{I}_3\ \mathbf{0}_3]$. Its normal $\mathbf{r}_1$ can therefore be computed as $\mathbf{r}_1 = (\mathbf{L}_1^\top + \mathbf{L}_2^\top)/\|\mathbf{L}_1^\top + \mathbf{L}_2^\top\|$. Finally, $\mathbf{r}_2$ is computed as the cross-product $\mathbf{r}_2 = \mathbf{r}_3 \times \mathbf{r}_1$.

We then proceed to initialize the position by estimating the tool origin $\mathbf{c}$. We parameterise it as $\mathbf{c} = \lambda(\mathbf{n} - \mathbf{m}) + \mathbf{n}$, where $\mathbf{n}, \mathbf{m} \in \mathbb{R}^3 \times \mathbb{R}^3$ represents any point-pair on the shaft axis. Concretely, $\mathbf{n}$ and $\mathbf{m}$ are picked at the intersection of plane π_1 with a second plane denoted π_2, defined as the plane through the shaft axis with normal $\mathbf{r}_2$. In other words, $\mathbf{n}$ and $\mathbf{m}$ are chosen in the nullspace of matrix $[\pi_1\ \pi_2] \in \mathbb{R}^{4\times 2}$, computable using an SVD. Plane π_2 is given by $\pi_2 = [\mathbf{r}_2^\top, R/\beta]^\top$ where $\beta = \|\mathbf{r}_1 \times \mathbf{L}_1\|/\|\mathbf{L}_1\|$. The problem is then to estimate $\lambda \in \mathbb{R}$, which resembles the problem of single-view point-on-line triangulation described in (Bartoli and Lapresté, 2008), for which we give a closed-form solution. The primary constraint we use is that there must be a point on the circle $\mathcal{C}$ which projects to the extracted image tool-tip $\mathbf{p}$. In other words, we constrain the circle center $\mathbf{c}$ by searching for the circle point projecting to $\mathbf{p}$. This circle point is not arbitrary, as it must be the closest to the camera. By using the known cylinder radius R, and recalling that $\mathbf{r}_2$ is oriented toward the camera, this point is thus given by $\mathbf{c} + R\mathbf{r}_2 = \lambda(\mathbf{n} - \mathbf{m}) + \mathbf{n} + R\mathbf{r}_2$. With this parameterisation, we can proceed to minimise the reprojection error with respect to $\mathbf{p}$, via the following least-squares problem:

$$\lambda = \arg\min_{\lambda \in \mathbb{R}} \left\| \mathcal{P}_u(\lambda(\mathbf{n} - \mathbf{m}) + \mathbf{n} + R\mathbf{r}_2) - \mathbf{p} \right\|^2 ,$$

where $\mathcal{P}_u$ defines the projection operator so that $\mathcal{P}_u([x, y, z]^\top) = [x/z, y/z]^\top$. This is a nonlinear least-squares problem, but because it involves only a single unknown λ, it has a simple closed-form solution. Defining $\mathbf{a} = \mathbf{n} - \mathbf{m}$ and $\mathbf{b} = \mathbf{n} + R\mathbf{r}_2$, and expanding the cost, we obtain:

$$\lambda = \arg\min_{\lambda \in \mathbb{R}} \left(\frac{\lambda a_x + b_x}{\lambda a_z + b_z} - p_x \right)^2 + \left(\frac{\lambda a_y + b_y}{\lambda a_z + b_z} - p_y \right)^2 .$$

By differentiating the cost, setting the result to zero and solving, we obtain the closed-form:

$$\lambda = \frac{a_z \left(b_x^2 + b_y^2 \right) - b_z \left(a_x b_x + a_y b_y \right)}{b_z \left(a_x^2 + a_y^2 \right) - a_z \left(a_x b_x + a_y b_y \right)}.$$

3.3.3. Refinement of 3D Pose

The refinement of the 3D pose is presented on the right in figure 6. The initial pose estimate $[\mathsf{R}\ \mathbf{c}]$ is refined using the geometric primitive maps $\mathcal{I}_*$, with $* \in \{el, ml, tt\}$. The pixel values of these maps give the truncated Euclidean distance to the tool shaft boundaries (edge-lines), to the image of the shaft axis (mid-line), and the image of the tool-tip $\mathbf{p}$. This refinement is accomplished by minimizing the pixel intensities along the geometric primitives predicted by the geometric model and its 3D pose. Specifically, we minimize the sum of the squared intensities as:

$$\begin{aligned}
\arg\min_{\mathbf{c},\mathsf{R}} \quad & \mathcal{I}_{tt}(\mathcal{P}(\mathbf{c} + R\mathbf{r}_2))^2 + \sum_j (\mathcal{I}_{el}(\mathbf{l}^j_{el,1})^2 + \mathcal{I}_{el}(\mathbf{l}^j_{el,2})^2 + \mathcal{I}_{ml}(\mathbf{l}^j_{ml})^2) \\
\text{with} \quad & \mathbf{l}^j_{el,1} = \mathcal{P}_c(\mathbf{c} - j\delta\mathbf{r}_3 + R(\cos(\gamma)\mathbf{r}_2 + \sin(\gamma)\mathbf{r}_1)), \\
& \mathbf{l}^j_{el,2} = \mathcal{P}_c(\mathbf{c} - j\delta\mathbf{r}_3 + R(\cos(\gamma)\mathbf{r}_2 - \sin(\gamma)\mathbf{r}_1)) \;\; \text{and} \\
& \mathbf{l}^j_{ml} = \mathcal{P}_c(\mathbf{c} - j\delta\mathbf{r}_3),
\end{aligned}$$

where $\mathcal{P}_c$ defines the projection operator such that for $\mathbf{Q} \in \mathbb{R}^3$, $\mathcal{P}_c(\mathbf{Q}) = \mathcal{P}_u(\mathsf{K}\mathbf{Q})$. The 2D point coordinates $\mathbf{l}^j_{el,*}$ and $\mathbf{l}^j_{ml}$ are the projection of 3D points regularly distributed along with the back-projection rays of the shaft boundaries and along the shaft axis, respectively. We define $\sin(\gamma) = \sqrt{1 - \cos(\gamma)^2}$ and $\cos(\gamma) = R/d$, where $d = \|[O_x, O_y]\|$ is the distance between the shaft axis and the camera optical center $[O_x, O_y, O_z]^\top = -\mathsf{R}^\top \mathbf{c}$. The step size is set to $\delta = 1mm$ and the step index varies in $j \in [1, 20]$. Any 3D point projected outside the image boundaries is assigned to a constant cost. The optimization is performed using the Levenberg-Marquardt algorithm (Moré, 1978), where we parameterize the rotation using Euler angles. The entire 3D pose estimation algorithm is described in table 1.

3.4. Datasets

We use two datasets. The first one is from the Endoscopic Vision Challenge (EndoVis Instrument Segmentation and Tracking sub-challenge), which contains two sub-datasets, namely robotic (articulated) and non-robotic (non-articulated). The robotic sub-dataset is a collection of 9050 images from 6 one minute laparoscopic videos of a large articulated needle driver tool used in an ex-vivo setup. The first 45 seconds of 4 videos consist of 4500 images used for training. The last 15 seconds belong to the testing dataset, which also includes the 2 other videos, leading to a total of 4550 testing images. The non-robotic sub-dataset is a collection of 300 images with 160 training images extracted from 4 laparoscopic colorectal surgeries (4×40), and 140 testing images from 6 laparoscopic surgeries ($4 \times 10 + 2 \times 50$). These two sub-datasets already include the binary segmentation masks. The geometric primitive annotations were added for images containing non-articulated tools, namely the non-robotic sub-dataset.

The second dataset is our proposed annotated data, which is non-robotic. We annotated the tool presence, the segmentation masks, and the geometric primitives for 635 laparoscopy images (see figure 7). We annotated the tool presence for another set of 3000 images, namely 1500 positive and 1500 negative images, respectively, for which some positive images contain multiple tools, therefore not usable for the geometric primitive extraction. These images are from 29 laparoscopic hysterectomy videos from the gynecology department of CHU Estaing

Table 1: The proposed tool-generic 3D pose estimation algorithm.

<u>OBJECTIVE</u> :
Given the geometric primitive maps $\mathcal{I}_{el}$, $\mathcal{I}_{ml}$ and $\mathcal{I}_{tt}$ and the tool shaft radius R, compute the tool 3D pose $[\mathsf{R}\ \mathbf{c}]$.
<u>ALGORITHM</u> :

1. Binarise $\mathcal{I}_{el}$ and $\mathcal{I}_{ml}$ using thresholding:

$$\mathcal{B}_{el} \leftarrow \text{threshold}(\mathcal{I}_{el}) \qquad \mathcal{B}_{ml} \leftarrow \text{threshold}(\mathcal{I}_{ml})$$

2. Compute $\mathbf{L}_1$, $\mathbf{L}_2$ and $\mathbf{L}_3$ using hough line detection applied on the binarised maps:

$$(\mathbf{L}_1, \mathbf{L}_2) \leftarrow \text{houghLine}(\mathcal{B}_{el}) \qquad \mathbf{L}_3 \leftarrow \text{houghLine}(\mathcal{B}_{ml}) \qquad \mathbf{L}_j \leftarrow \mathbf{L}_j / \sqrt{L_{1j}^2 + L_{2j}^2}$$

3. Extract the imaged tool-tip: $\mathbf{p} \leftarrow \arg\max_{x,y} \mathcal{I}_{tt}(x, y)$

4. Initialise the tool orientation $\mathsf{R} = [\mathbf{r}_1\, \mathbf{r}_2\, \mathbf{r}_3]$:

$$\mathbf{r}_3 \leftarrow \arg \min_{\substack{\mathbf{X} \in \mathbb{R}^3 \\ \|\mathbf{X}\|=1}} \|\mathsf{A}\mathbf{X}\|^2 \quad \text{where} \quad \mathsf{A} = \begin{bmatrix} \mathbf{L}_1^\top \\ \mathbf{L}_2^\top \\ \mathbf{L}_3^\top \end{bmatrix}$$

$$\mathbf{r}_1 \leftarrow (\mathbf{L}_1^\top + \mathbf{L}_2^\top)/\|\mathbf{L}_1 + \mathbf{L}_2\| \qquad \mathbf{r}_2 \leftarrow \mathbf{r}_3 \times \mathbf{r}_1$$

5. Initialise the tool position $\mathbf{c}$:

$$\pi_1 \leftarrow [\mathbf{L}_1^\top + \mathbf{L}_2^\top\ 0]^\top \quad \beta \leftarrow \|\mathbf{r}_1 \times \mathbf{L}_1\| / \|\mathbf{L}_1\| \quad \pi_2 \leftarrow [\mathbf{r}_2^\top\ R/\beta]^\top \quad (\mathsf{U}, \Sigma, \mathsf{V}) \leftarrow \text{svd}([\pi_1\, \pi_2])$$

$$\mathbf{n} \leftarrow \mathsf{V}(:, 3) \qquad \mathbf{m} \leftarrow \mathsf{V}(:, 4) \qquad \mathbf{a} \leftarrow \mathbf{n} - \mathbf{m} \qquad \mathbf{b} \leftarrow \mathbf{n} + R\mathbf{r}_2$$

$$\lambda \leftarrow \frac{a_z\left(b_x^2 + b_y^2\right) - b_z\left(a_x b_x + a_y b_y\right)}{b_z\left(a_x^2 + a_y^2\right) - a_z\left(a_x b_x + a_y b_y\right)} \qquad \mathbf{c} \leftarrow \lambda(\mathbf{n} - \mathbf{m}) + \mathbf{n}$$

6. Refine the pose by minimising the sum of the squared intensities as:

$$\begin{aligned} (\mathsf{R}, \mathbf{c}) \leftarrow \quad & \arg\min_{\mathbf{c},\mathsf{R}} \ \mathcal{I}_{tt}(\mathcal{P}_c(\mathbf{c} + R\mathbf{r}_2))^2 + \sum_{j=1}^{20} (\mathcal{I}_{el}(\mathbf{l}_{el,1}^j)^2 + \mathcal{I}_{el}(\mathbf{l}_{el,2}^j)^2 + \mathcal{I}_{ml}(\mathbf{l}_{ml}^j)^2) \\ \text{with} \quad & \mathbf{l}_{el,1}^j = \mathcal{P}_c(\mathbf{c} - j\delta\mathbf{r}_3 + R(\cos(\gamma)\mathbf{r}_2 + \sin(\gamma)\mathbf{r}_1)), \\ & \mathbf{l}_{el,2}^j = \mathcal{P}_c(\mathbf{c} - j\delta\mathbf{r}_3 + R(\cos(\gamma)\mathbf{r}_2 - \sin(\gamma)\mathbf{r}_1)), \\ & \mathbf{l}_{ml}^j = \mathcal{P}_c(\mathbf{c} - j\delta\mathbf{r}_3) \ \text{ and } \ \delta = 1. \end{aligned}$$

(Clermont-Ferrand, France), which were randomly split into train and test sets patient-wise. We used 1016 images for tool detection, namely 508 positives and 508 negatives, for training, and 3254 images, namely 1627 positives and 1627 negatives, for testing. If the tool shaft is not visible at all, the image is marked as negative and corresponds to cases for which the pose cannot (and, hence, must not) be computed. When a small part of the tool shaft is visible, the image is marked as positive. For segmentation and geometric primitive extraction, we used 508 and 127 images for training and testing, respectively. The distribution in terms of image disturbances of the 935 images composing EndoVis non-robotic and our dataset is as follows. No disturbance: 67 % (630 images); motion blur: 10 % (97 images); presence of trocar: 4 % (33 images); bleeding: 7 % (65 images); smog: 7 % (63 images); tool occlusion: 5 % (47 images). Table 2 shows a summary of the datasets where the numbers of annotations for detection, segmentation, and geometric primitive extraction are reported.

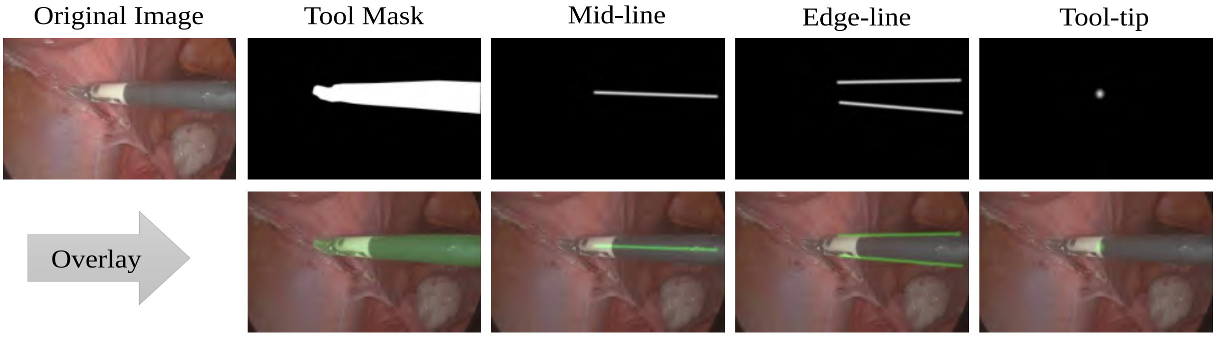


Figure 7: Example of an annotated image. We used ImageJ (Schneider et al., 2012) and basic image processing, as described in GitHub (see Hasan et al., 2020).

Table 2: Statistics of the datasets used for the quantitative evaluations of the detection, segmentation, and geometric primitive extraction. Pos. and Neg. stand for positive and negative images, respectively.

Source and usage			Tool mask	Edge-line	Mid-line	Tool-tip	Detection (Pos./Neg.)
EndoVis	Robotic (Articulated)	Train	4,500	-	-	-	-
		Test	4,550	-	-	-	-
	Non-robotic (Non-articulated)	Train	160	160	160	160	160 (160/0)
		Test	140	140	140	140	140 (140/0)
Our proposed data (Non-articulated)		Train	508	508	508	508	1,016 (508/508)
		Test	127	127	127	127	254 (127/127)
			-	-		-	3,000 (1,500/1,500)

3.5. Loss Function, Training Strategy, and Evaluation Criteria

3.5.1. Loss Function

The binary or categorical cross-entropy (CE) functions are widely used as loss function in both classification and semantic segmentation in CNN training. However, they may lead to biases as a surgical tool's area is significantly smaller than the area of the background (Garcia-Peraza-Herrera et al., 2017). Hence, we chose the sum of *binary CE* and *IoU* as loss function L_{seg}:

$$\begin{aligned} L_{seg}(y, \hat{y}) = & -\frac{1}{N}\sum_{i=1}^{N}(y_i \log \hat{y}_i + (1 - y_i)\log(1 - \hat{y}_i)) \\ & +1 - \frac{\sum_{i=1}^{N} y_i\hat{y}_i}{\sum_{i=1}^{N} y_i + \sum_{i=1}^{N} \hat{y}_i - \sum_{i=1}^{N} y_i\hat{y}_i}, \end{aligned} \tag{1}$$

where y and $\hat{y}$ are the true label and predicted probability, respectively. The loss function for the detection and regression sub-networks are *CE* and *mean squared error*, respectively. The total loss function L of ART-Net is the weighted sum of the individual loss function of each sub-network:

$$L = L_{det}(y_d, \hat{y}_d) + L_{seg}(y_s, \hat{y}_s) + L_{mid}(y_{ml}, \hat{y}_{ml}) + L_{edge}(y_{el}, \hat{y}_{el}) + L_{tip}(y_{tt}, \hat{y}_{tt}), \tag{2}$$

where y_d, y_s, y_{ml}, y_{el}, and y_{tt} are the true labels and $\hat{y}_d$, $\hat{y}_s$, $\hat{y}_{ml}$, $\hat{y}_{el}$, and $\hat{y}_{tt}$ are the predicted probabilities for detection, segmentation, mid-line, edge-line, and tool-tip sub-networks, respectively. The loss function L is optimised using *adadelta* (Zeiler, 2012) with *initial learning rate* = 1.0 and *decay factor* = 0.95.

3.5.2. Training Strategy

In the proposed strategy, the encoder's kernels are initialized using the pre-trained weights from ImageNet (Deng et al., 2009), and the kernels of the decoders are initialized using the 'glorot uniform' distribution. Two stages of training and testing were used (see section 4.2 for details). In the first stage, referred to as *stage-1*, we trained and tested only the segmentation sub-network of ART-Net on the EndoVis (robotic) dataset. In the second stage, referred to as *stage-2*, we trained and tested the whole ART-Net on the combined EndoVis (non-robotic) and our annotated data.

3.5.3. Evaluation

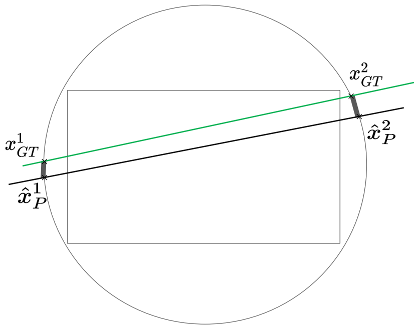


Figure 8: Illustration of the error from equation (3) used for the evaluation of geometric primitive extraction for the tool edge-lines and mid-line. The arc's length is defined by the intersection of the unit circle with the ground-truth and predicted lines referred to as x^i_{GT} and $\hat{x}^i_P$, respectively, is measured for the two intersection point-pairs. Given one predicted line, the reported error is the average of these two values.

Tool detection was evaluated using average precision and average accuracy. Tool segmentation was evaluated using mean Dice Similarity Coefficient (mDSC), mean Intersection over Union (mIoU), mean Sensitivity (mSn), and mean Specificity (mSp). mDSC and mIoU quantify the percentage overlap between the true and the predicted tool masks. mSn and mSp quantify the false-positive rate and false-negative rate, respectively. The predicted edge and mid-line primitives were quantitatively evaluated using the mean Arc Length (mAL) error between the true x_{GT} and the predicted $\hat{x}_P$ points, as shown in figure 8:

$$mAL = \frac{1}{N}\sum_{i=1}^{N}\frac{d(x^1_{i,GT}, \hat{x}^1_{i,P}) + d(x^2_{i,GT}, \hat{x}^2_{i,P})}{2}, \tag{3}$$

where d is the arc length of the unit circle and N is the total number of images. The predicted tool-tip is evaluated by its Euclidean distance to the true tip point.

4. Experimental Results

4.1. Tool Detection

A tool's presence is defined as a positive case, whereas the absence of a tool is defined as a negative case. The proposed ART-Net successfully detects the tool presence in all the positive images, hence obtaining an average precision and accuracy both of 100.0 %. The use of GAP for vectorizing the 2D feature maps into a single long continuous vector was compared against traditional flatten layers. ART-Net fails to detect the tool for 311 positive images with traditional flatten layers, leading to an average precision and accuracy of 87.4 %, and 89.6 %, respectively. Moreover, ART-Net with GAP is much more compact, with $17M$ parameters, than its implementation with flattening layers, comprising $42M$ parameters. A few qualitative results on positive and negative images are displayed in figure 9. This shows that the proposed detection sub-network can successfully identify the tool's presence or absence in challenging conditions. For instance, the trocar (1^{st} row - 6^{th} column and 2^{nd} row - 7^{th} column) and operating room (1^{st} row - 8^{th} column) are successfully detected as negatives.

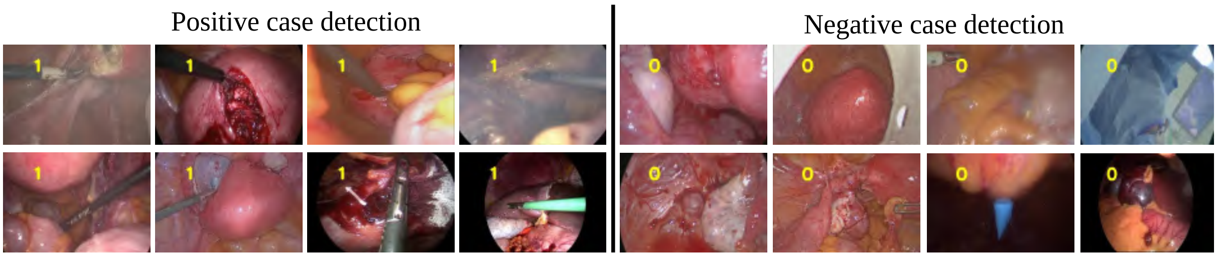


Figure 9: Qualitative results for surgical tool detection. The left and right four columns are respectively for the positive and negative cases. The yellow tag on the image is the tool detection flag (1 is tool presence, and 0 is tool absence). More qualitative results for detection are available in GitHub (Hasan et al., 2020).

4.2. Tool Segmentation

Several state-of-the-art methods for tool segmentation (Garcia-Peraza-Herrera et al., 2017, 2016; Pakhomov et al., 2019) evaluate their methods on the EndoVis robotic sub-dataset, which comprises a very large number of images compared to the non-robotic one. Laina et al. (2017); Milletari et al. (2018) do not share their implementations, which prevents us from comparing ART-Net against their methods on the EndoVis non-robotic and the proposed sub-datasets. In order to increase the possible comparisons, we, therefore, propose a preliminary training and testing stage of ART-Net, referred to as stage-1, over the EndoVis robotic dataset. Specifically, in stage-1, ART-Net is reduced to its segmentation sub-network and the loss function to the L_{seg} term only in equation (1). Importantly, stage-1 also allows us to pre-train ART-Net on a large laparoscopic image dataset. A second training and testing stage, referred to as stage-2, was then performed on the non-robotic laparoscopic images, namely images associated with a tool model presenting a cylindrical shaft, as considered in our generic method. Stage-2 uses the full loss function L in equation (2), which includes the detection, segmentation, and geometric primitive extraction terms. An evaluation of ART-Net for segmentation in stage-2 was possible but against fewer state-of-the-art methods than in stage-1, namely against several instances of U-Net, FCN8, and ART-Net with several ablations.

Results from stage-1. We report the segmentation results using several scores and compare ART-Net against state-of-the-art methods. U-Net and FCN8s were trained and tested on the same dataset, namely EndoVis (robotic), and optimized using the same loss as for ART-Net, namely the cross-entropy term L_{seg} in equation (1). The obtained quantitative and qualitative results, without any post-processing, are shown in table 3 and figure 11, respectively. ART-Net is very close to the winning method for most of the criteria, namely 0.2 pp for mDSC and 0.4 pp for mean specificity[2], and wins for the mean IoU together with CFCM, which comprises about twice as many parameters as ART-Net. ART-Net's results represent a very good compromise between specificity and sensitivity, reaching a specificity very close to the winning one while maintaining an average sensitivity. The use of FrG in the proposed ART-Net improves all the scores but mean sensitivity. The effect of its use on the segmentation masks is bestowed for several laparoscopic images in figure 10. The proposed ART-Net without FrG corresponds to a U-Net with separable convolutions in the decoder. The use of separable convolutions instead of traditional convolutions improves

[2] pp stands for 'percentage points' and represents the unit used for the difference of two percentages

Table 3: Quantitative metrics for the segmented tool masks from ART-Net, U-Net, FCN8s, and state-of-the-art segmentation networks on EndoVis-2015 (robotic). The metrics were computed using the true labels and semantic labels obtained from the networks. Best results are in bold, second-best underlined, third-best underlined twice.

Networks	Params	Pre-train	Experiments	Metrics			
				mDSC	mSn	mSp	mIoU
FCN8s (Garcia-Peraza-Herrera et al., 2016)	$134M$	PASCAL-context	–	78.8 %	72.2 %	95.2 %	70.9 %
CSL (Laina et al., 2017)	$31M$	ImageNet	–	88.9 %	86.2 %	**99.0** %	80.0 %
CFCM (Milletari et al., 2018)	$33M$	–	–	**89.5** %	88.8 %	98.8 %	**81.0** %
ToolNetH (Garcia-Peraza-Herrera et al., 2017)	$7.4M$	–	–	82.2 %	–	–	74.4 %
ToolNetMS (Garcia-Peraza-Herrera et al., 2017)	$7.3M$	–	–	80.4 %	–	–	72.5 %
FCN (Pakhomov et al., 2019)	$23M$	PASCAL VOC	–	87.4 %	85.7 %	98.8 %	77.6 %
U-Net	$38M$	ImageNet	CE loss	77.3 %	**94.7** %	95.3 %	64.1 %
			PL*	87.5 %	93.5 %	97.5 %	78.1 %
FCN8s	$134M$	ImageNet	CE loss	69.9 %	89.0 %	94.2 %	55.1 %
			PL*	86.4 %	85.9 %	98.3 %	76.5 %
ART-Net	$17M$	ImageNet	CE loss	82.0 %	88.7 %	96.6 %	70.0 %
	$17M$		No FrG	86.9 %	92.9 %	97.4 %	77.1 %
	$38M$		No SC*	87.3 %	93.5 %	97.3 %	78.0 %
ART-Net (Proposed, 2020)	$17M$	ImageNet	PL* + SC* + FrG	89.3 %	88.1 %	98.6 %	**81.0** %
Differences between proposed ART-Net and winner				0.2 pp	6.6 pp	0.4 pp	0.0 pp

*SC: Depth-wise separable convolution, and PL: Proposed loss function.

both the mDSC and mIoU by the margins of 2.9 pp and 3.5 pp respectively and reduces the numbers of parameters from $38M$ to $17M$.

The qualitative results of the segmented masks (see figures 10 and 11) show that the masks generated by FCN8s are coarse at the tool boundary, with checkerboard artifacts and more false positives (FP). The U-Net model obtains better segmentation masks than FCN8s, both qualitatively and qualitatively. However, ART-Net with the proposed FrG connection outperforms. This shows that the proposed segmentation sub-network provides better segmentation masks, even for challenging images, than U-Net, FCN8s, and the other competing networks.

Results from stage-2. We evaluate the segmentation results of *stage-2* based on EndoVis (non-robotic) and our annotated data, which are non-articulated. The proposed ART-Net, including ablations, FCN8s, and U-Net, are evaluated. The loss function of equation (1), referred to as proposed loss, was used in this evaluation, as it was shown to perform best in stage-1. The results are reported in table 4. ART-Net wins for mDSC, mSn, and mIoU. It is incredibly close (0.1 pp) to the winning method, namely U-Net, for mSp. A qualitative evaluation is exhibited in figure 12, where ART-Net with FrG outperforms. Additional results in case of tool occlusion are shown in figure A.22, where segmentation results with multiple tools seen simultaneously are also provided.

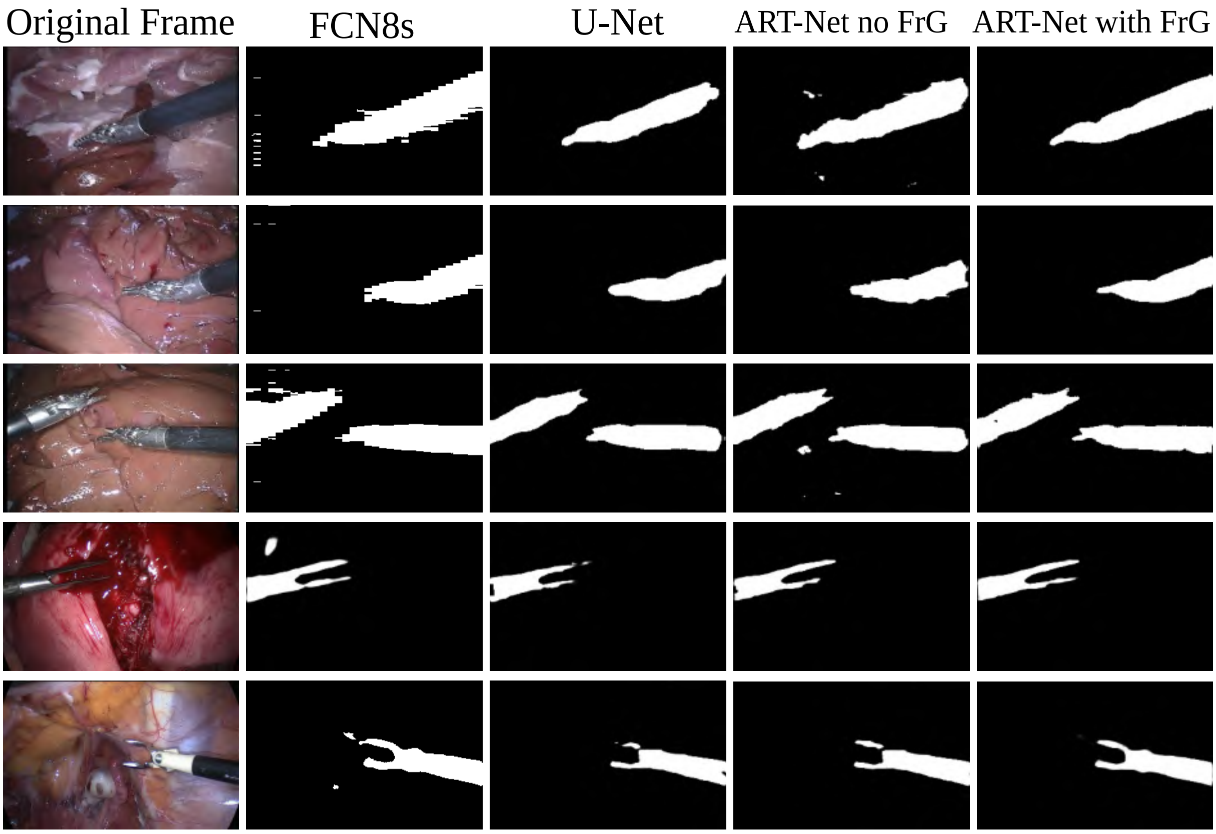


Figure 10: Examples of segmentation results from FCN8s, U-Net, ART-Net without FrG, and with FrG. The input images for the first three rows are from the EndoVis robotic dataset and from our proposed dataset for the last two rows.

Table 4: Quantitative evaluation of segmentation for stage-2. Several U-Net instances, including ablations, U-Net, and FCN8s, are evaluated on EndoVis (non-robotic) and our annotated data. Best results are in bold, second-best underlined, and third-best underlined twice.

Networks	Params	Experiments	Metrics			
			mDSC	mSn	mSp	mIoU
U-Net	$38M$	–	$\underline{92.1\,\%}$	$\underline{92.8\,\%}$	$\mathbf{99.1}\,\%$	$\underline{86.7\,\%}$
FCN8s	$134M$	–	87.8 %	86.3 %	$\underline{99.0\,\%}$	79.1 %
	$17M$	CE loss	87.3 %	91.2 %	98.2 %	78.7 %
ART-Net	$17M$	No FrG	87.0 %	89.5 %	$\underline{\underline{98.3\,\%}}$	78.6 %
	$38M$	No SC*	$\underline{\underline{91.8\,\%}}$	$\underline{\underline{92.6\,\%}}$	$\underline{99.0\,\%}$	$\underline{\underline{86.2\,\%}}$
ART-Net	$17M$	Proposed loss + SC* + FrG	$\mathbf{93.2}\,\%$	$\mathbf{95.3}\,\%$	$\underline{99.0\,\%}$	$\mathbf{88.2}\,\%$
Differences between proposed ART-Net and winner			0.0 pp	0.0 pp	0.1 pp	0.0 pp

*SC: Depth-wise separable convolution

4.3. Geometric Primitive Extraction

We present geometric primitive extraction results from the proposed ART-Net and the U-Net with different loss functions for the regression task and ablation studies. We drop

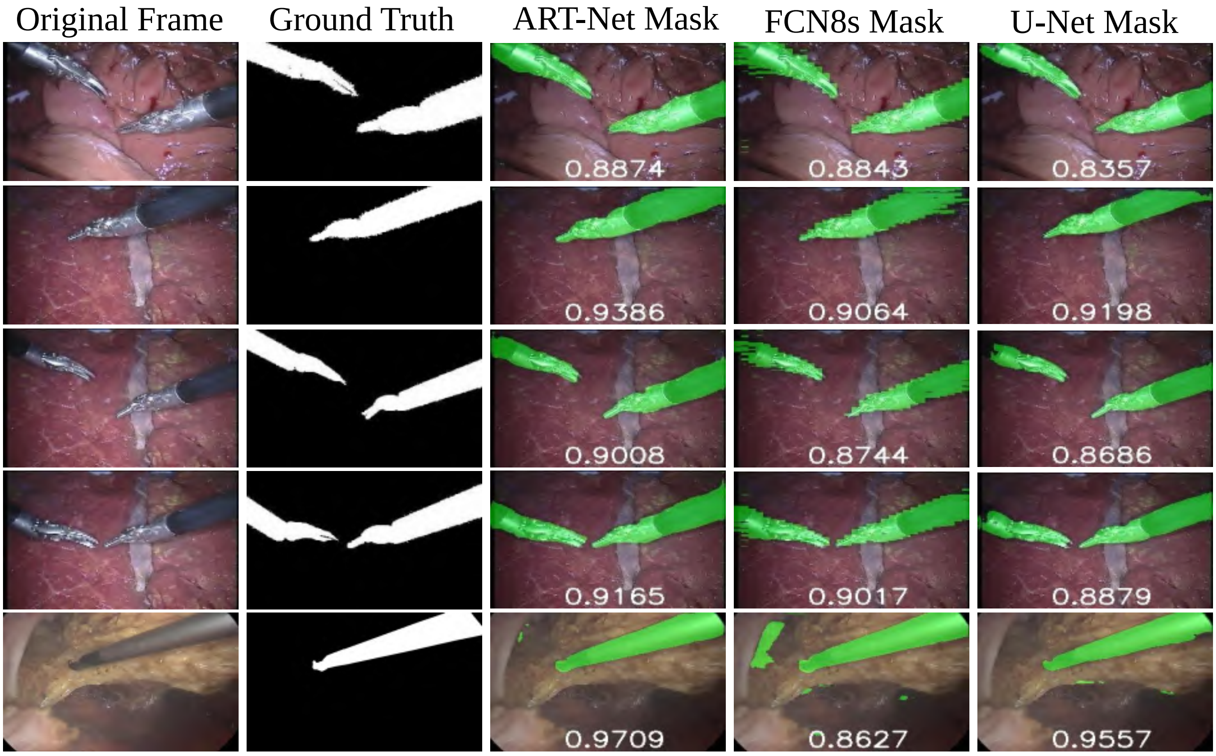


Figure 11: Segmentation results at *stage-1* from ART-Net, FCN8s, and U-Net. The segmentation masks are shown in green along with the DSC values. More segmentation results are available in GitHub (Hasan et al., 2020). The first four rows' input images are from the EndoVis robotic dataset and from the EndoVis (non-robotic) dataset for the last row.

FCN8s from the experiments as it produces coarse and zigzag tool boundaries, which is undesirable as a geometric primitive for 3D pose estimation. Additionally, the SIMO ART-Net, with FCN8s structure, has approximately $435M$ parameters, which is overly expensive to train. The predicted geometric primitives were evaluated quantitatively using the mean and median AL values in degrees, from section 3.5, and the Euclidean distance in pixels. Quantitative and qualitative results for the predicted geometric primitives are shown in table 5 and figure 13, respectively. Table 5 shows that the proposed ART-Net, with L_2 loss function, produces the best results for geometric primitive extraction with mAL and medAL of 2.45°, 1.71° and 2.23°, 1.34° respectively for the edge-lines and mid-line, as well as mED and medED of 9.3 and 3.2 pixels for the tool-tip. For both networks, the L_2 loss outperforms the other two-loss functions for all the primitives. The other two-loss functions defeat the L_1 loss. It also has the drawback not having a continuous derivative. The proposed ART-Net

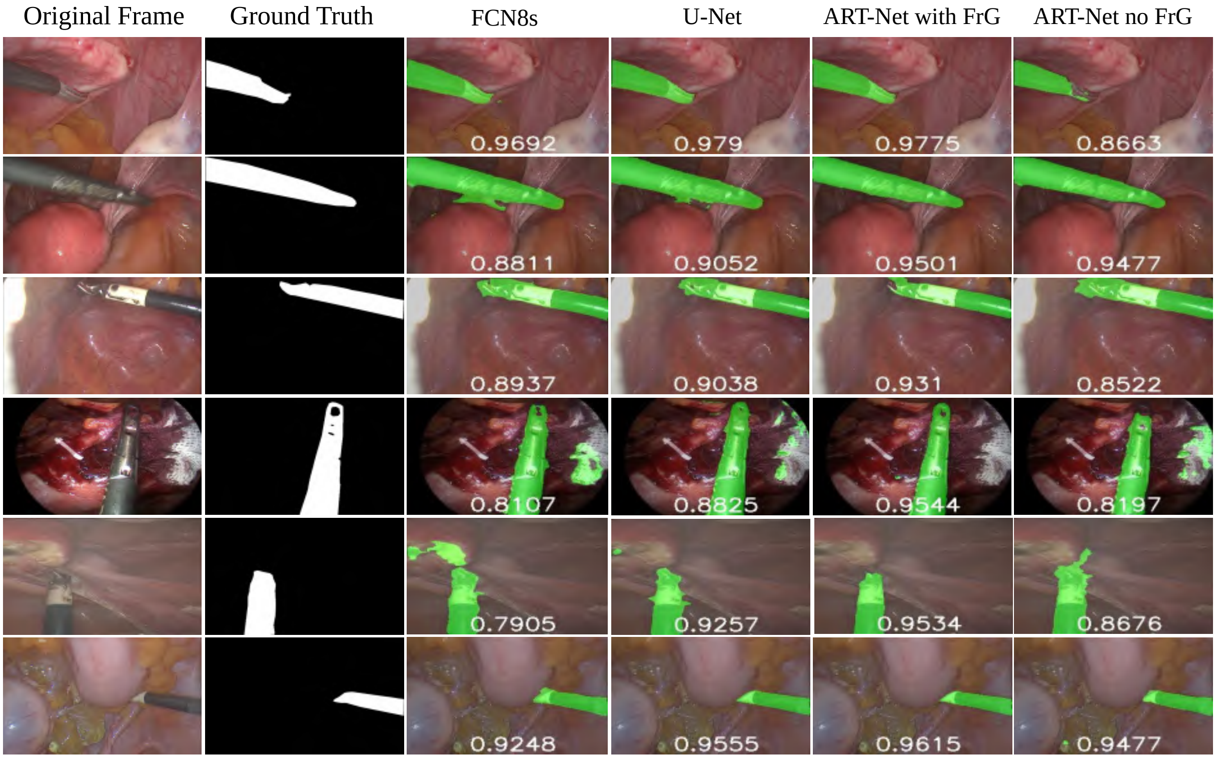


Figure 12: Segmentation results at *stage-2* applied on the combined EndoVis (non-robotic) and our annotated dataset. The segmentation masks are shown in green along with the DSC values. From top to bottom, the input images are without disturbance, with motion blur, presence of trocar, bleeding, smog, and tool occlusion. More segmentation results are available in GitHub (Hasan et al., 2020).

Table 5: Experimental results for geometric primitive extraction from different networks and loss functions, where we have reported mean AL (mAL) and median AL (medAL) for edge-line and mid-line, mean ED (mED), and median ED (medED) for tool-tip. Best results are in bold, second-best underlined.

Networks	Params	Experiments	Geometric primitives and Metric					
			Edge-line		Mid-line		Tool-tip	
			mAL	medAL	mAL	medAL	mED	medED
U-Net	38M	Mean Squared Error (L_2 loss)	2.66°	**1.49°**	4.30°	1.99°	15.60	5.40
		Mean Absolute Error (L_1 loss)	2.97°	1.65°	4.53°	1.94°	67.20	20.70
		Huber loss	2.62°	1.53°	3.68°	2.00°	22.01	7.10
ART-Net	17M	Mean Squared Error (L_2 loss)	**2.45°**	1.71°	**2.23°**	**1.34°**	**9.30**	**3.20**
		Mean Absolute Error (L_1 loss)	2.96°	1.64°	4.30°	1.89°	65.80	16.60
		Huber loss	2.62°	1.62°	3.95°	2.07°	14.80	5.90
ART-Net	17M	No FrG	4.29°	1.63°	4.89°	3.04°	22.04	8.74
	38M	No separable convolution (SC)	2.76°	1.79°	3.93°	2.99°	14.76	8.49
ART-Net	17M	SC + FrG + L_2 loss	**2.45°**	1.71°	**2.23°**	**1.34°**	**9.30**	**3.20**
Differences between proposed ART-Net and winner			0.0°	0.22°	0.0°	0.0°	0.0	0.0

*SC: Depth-wise separable convolution

with the L_2 loss outperforms the best results of the U-Net for all the primitives, as shown in table 5. The medAL metric for ART-Net with the L_2 loss shows that the error for 50 % of the lines lies below 1.71° and 1.34°, respectively, for the edge-lines and mid-line. The mean and median value of Euclidean distance, in pixels, between the true and predicted tool-tip, also shows the success of tool-tip detection by ART-Net. The qualitative results, in figure 13, show that the predicted edge-lines and mid-line almost overlap entirely with the ground-truth. The qualitative and quantitative results of geometric primitive extraction both validate the excellent performance of the regression sub-networks of the proposed ART-Net.

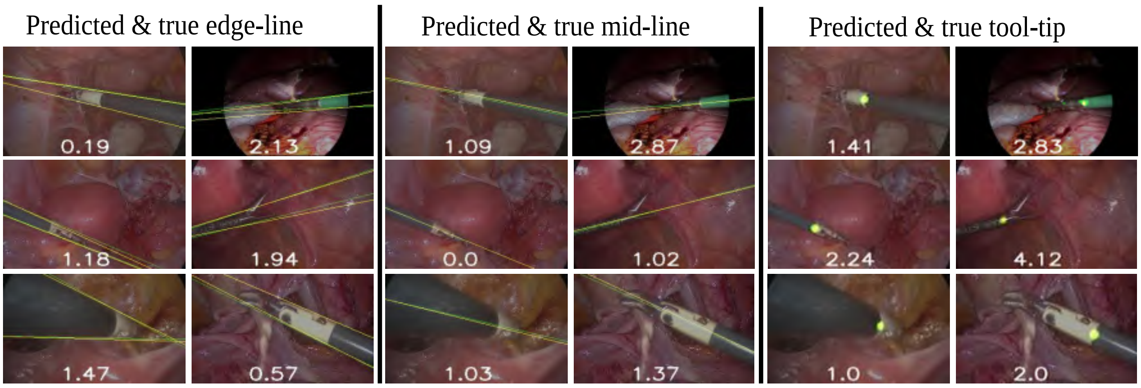

Predicted & true edge-line
Predicted & true mid-line
Predicted & true tool-tip
0.19
2.13
1.09
2.87
1.41
2.83
1.18
1.94
0.0
1.02
2.24
4.12
1.47
0.57
1.03
1.37
1.0
2.0

Figure 13: Geometric primitives extracted from the geometric primitive maps delivered by ART-Net (in green) and their ground-truth (in yellow) overlaid on the input images. The arc length measured in degrees and Euclidean distance measured in pixels are also overlaid. Additional results of geometric primitive extraction are available in GitHub (Hasan et al., 2020).

Quantitative results of ART-Net for each type of challenging conditions and each task are reported in table 6. ART-Net shows similar segmentation performance for all the perturbations but in the presence of smog and bleeding with a mIoU $6\,pp$ and $7\,pp$ lower than the mIoU over the full test set. It shows similar performance for mid-line and edge-line extractions for all the perturbations but in the presence of trocar and bleeding, with an error of about 5° higher than the one over the full test set. It shows similar performance for the tooltip extraction for all the perturbations but in the presence of bleeding, with an error of 11.8 pixels higher than the average error over the full test set.

4.4. 3D Pose Estimation

4.4.1. Qualitative Image-based Evaluation

The 3D pose of the surgical tool is estimated from the sets of geometric primitives, following the steps of section 3.3. The reprojection of the estimated 3D pose on the tool is

Table 6: Quantitative evaluation of ART-Net under challenging conditions. The results are reported for each type of challenging condition. They are reported in terms of mIoU, mAL, ED, and accuracy for tool segmentation, edge-line, mid-line, and tool-tip extraction, and tool detection, respectively.

Type of challenging conditions	Tool Segmentation	Geometric Primitives			Tool Detection
		Edge-line	Mid-line	Tool-tip	
No disturbance	90.1 %	1.92°	2.03°	7.4	100.0 %
Motion blur	88.5 %	2.67°	2.78°	11.4	100.0 %
Trocar presence	86.4 %	7.05°	7.79°	4.8	100.0 %
Bleeding	81.2 %	3.22°	4.54°	21.1	100.0 %
Smog	82.2 %	4.14°	2.72°	9.9	100.0 %
Tool occlusion	89.9 %	1.18°	0.84°	1.9	100.0 %
Full test set	88.2 %	2.45°	2.23°	9.3	100.0 %

conferred in figure 14 for qualitative assessment, where it is perceived that the estimated 3D pose leads to shallow reprojection errors despite tool type, color, and orientation. From figure 14, it is also noticed that the reprojection errors are low even if the tool diameter and head length vary significantly. Figure 15 shows that the estimated pose is precise even in the presence of motion blur between the surgical tool shaft and tool head (figure 15 (a)). We observe that first-order (figure 15 (b)) and second-order (figure 15 (c)) local filtering fail to provide gradient information at the boundary of the tool shaft and head, whereas the proposed CNN approach successfully localizes the tool-tip. We observe in figure 15 (d,e) that there is no gradient information along the tool edge boundary available from local Sobel filtering either, but that the proposed method can nonetheless localize the tool boundary and estimate 3D pose precisely. Figure 15 (d,e) also shows that non-linear refinement brings the contour of the surgical tool precisely at the maximum gradient of the tool's boundary,

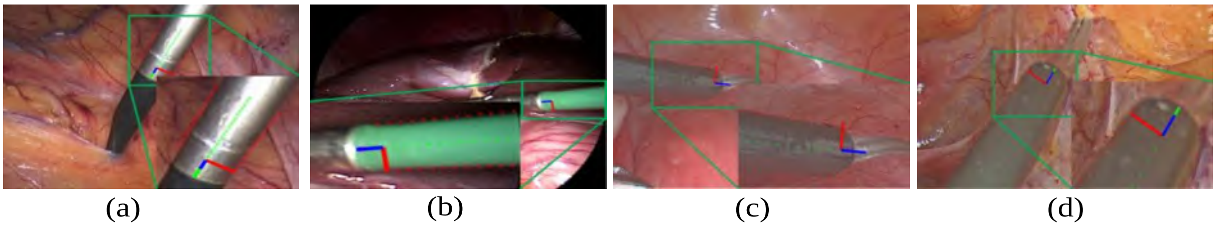


Figure 14: Examples of tool reprojection from the estimated poses by our framework. The red, green and blue lines show the direction of $\mathbf{r}_1$, $\mathbf{r}_2$ and $\mathbf{r}_3$, representing the estimated tool orientation. The red crosses lie on the boundaries of the shaft reprojection. The green crosses lie on the reprojection of the shaft axis. The images are purposefully associated with different navigation directions of the tool: (a) top to the bottom, (b) right to the left, (c) left to the right, and (d) bottom to the top.

which would be extremely hard to achieve using a local filtering method.

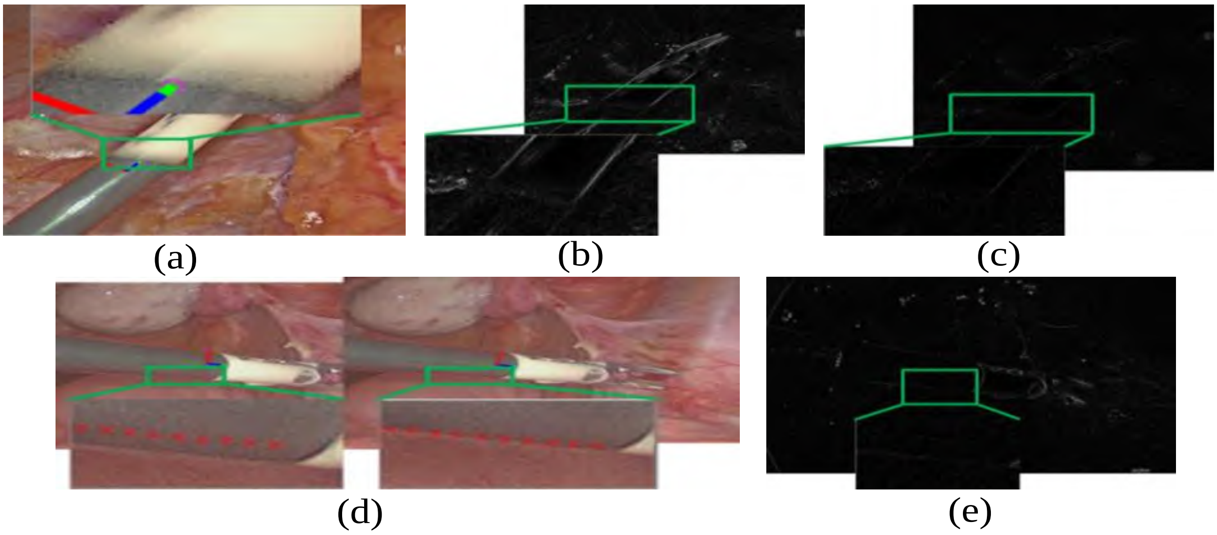


Figure 15: Two examples of challenging images where classical local filters fail to detect the imaged shaft boundaries. First row: (a) first example, for which the proposed framework precisely localizes the tool-tip (purple circle) in the presence of motion blur. The (b) Sobel and (c) Laplacian filter responses are extremely weak along the shaft boundary, therefore not usable for pose estimation. Second row: (d) reprojection of the shaft boundaries and axis in red and green, respectively, for initial (left) and refined (right), poses estimated based on the geometric primitive maps from ART-Net; (e) response of a Sobel filter, which happens to be extremely weak along the shaft boundary, showing that the use of such local filtering is not appropriate in this case.

4.4.2. Quantitative 3D Evaluation

Two experiments were conducted to perform a quantitative evaluation of 3D pose estimation from animal and patient data, respectively. For the first experiment, on animal data, the errors on the estimated tool 3D pose are reported in terms of mean angular errors between the estimated and true shaft orientations and mean absolute errors on the X, Y, and Z coordinates between the estimated and true positions of the tool origin and the tool-tip. For the second experiment, on data from laparoscopic surgery of the uterus, the errors on the estimated tool 3D pose are reported in terms of 3D Euclidean distance between the estimated tool-tip that physically touches the uterus and a preoperative 3D model of a uterus registered using (Collins et al., 2016) across the image collection. The tool radii R and tool-tip lengths h were measured using a numerical caliper. The tool 3D pose estimation method closest to ours is (Agustinos and Voros, 2015). However, this method's implementation is not publicly available and would be difficult to reproduce, preventing a direct experimental comparison. Nonetheless, this method uses tool boundaries detection, which is not trained end-to-end and uses local filtering instead of DNN. Hence, it is likely not to be triggered in challenging surgery conditions, and its robustness and performance will depend on the color and material of the tool shaft and tip.

Evaluation from animal data. Two videos of laparosurgery performed on a pig were collected. A hook scissor and curved dissecting forceps were used in the first and second videos, respectively. To obtain the ground-truth of the 3D pose, a chessboard was stuck on each of the tools in a position well visible on the laparoscopic videos, as shown in figure 16 (a). For each acquisition, a sub-image not containing the chessboard was manually extracted (shown in figure 16 (a) as the left-most, undimmed part) and used as an input image to ART-Net. An example of an estimated pose is shown in figure 16 (b). It is compared from two different viewpoints against the ground-truth computed from the chessboard in figure 16 (c-d). The ground-truth was computed as follows. A 3D reconstruction of the tool was performed using the Meshroom software (AliceVision, 2018) from a collection of photographs taken with a high-resolution camera. The chessboard corners were then manually selected and refined to reach subpixel accuracy. The tip points of the tools were also selected. The reconstructed cameras were used to triangulate the corner and tip points. The triangulated points were then refined through bundle adjustment using Matlab. The obtained 3D reconstruction was finally used to get the ground-truth of the tool 3D pose using EPnP (Lepetit et al., 2009) from 3D-2D correspondences manually selected in the laparoscopic images. 50 images were extracted from each video. As the image domain associated with animal data differs from the one related to the training data, namely patient data, a subset of images for which the pose initialization provided acceptable reprojection errors were kept for ground-truth computation and evaluation of the tool 3D pose. It led to the use of 17 images for the first video and 15 for the second one. The average mean absolute errors on the X, Y, and Z coordinates of the tool origin and the tool-tip. The shaft orientation's angular errors are reported in the table 7. Importantly, these errors represent an upper bound on the actual error, as only a partial region of the imaged tool shaft, namely the part not covered by the chessboard, is used to compute the pose, as shown in figure 16(a).

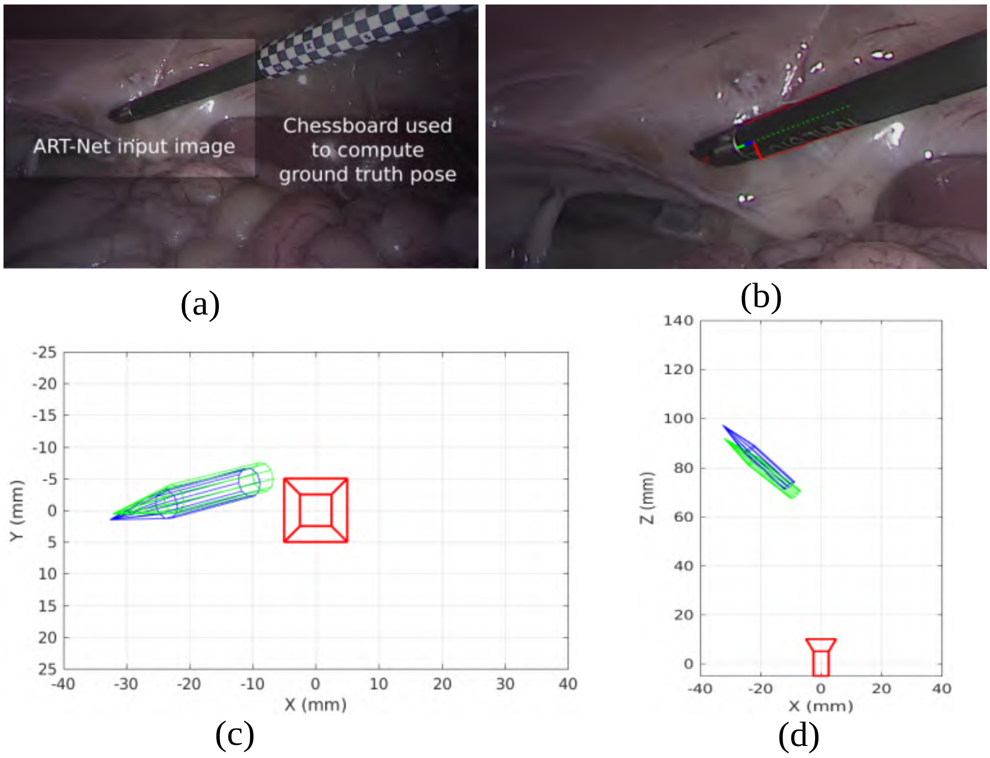


Figure 16: (a) General image for 3D pose evaluation from animal data. A sub-image not containing the chessboard is manually extracted (shown left as the undimmed part) and used as an ART-Net input image. The chessboard is used to compute the ground-truth 3D pose. (b) Estimated 3D pose reprojected on the ART-Net input image. (c) Orthographic view along the camera optical axis of the estimated (blue) and ground-truth (green) poses. (d) Top orthographic view of the estimated and ground-truth poses.

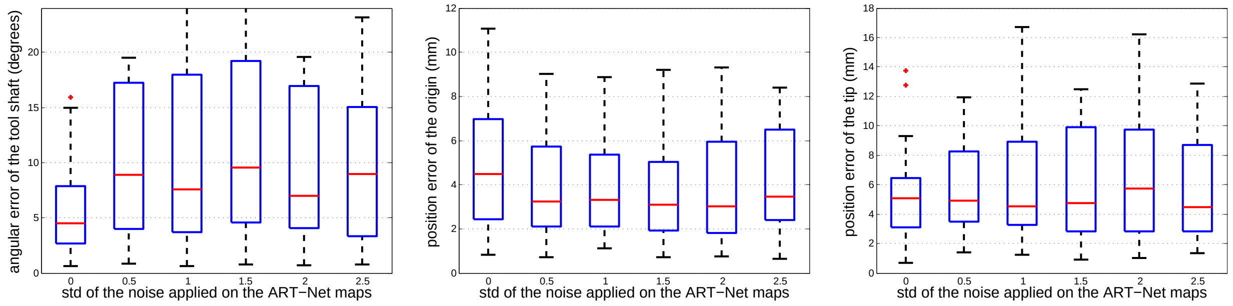
The two input model parameters, namely the tip length and the shaft radius, may be affected by measurement errors or manufacturing inaccuracies, possibly leading to an increase in pose estimation error. An error δ on the tooltip length does not affect the estimated shaft position and orientation but induces an error of magnitude δ on the estimated tooltip position. The effect of an error on the shaft radius is more complex to figure out analytically. We hence propose to evaluate it numerically. Specifically, we report tool 3D pose estimation results with an input shaft radius perturbed by errors of 1% and 5%, for the scissor tool. These represent large errors, given the high precision manufacturing of surgical tools required by the use of trocars. These results are reported in table 7. The pose error is stable for the 1% perturbation case: compared to the results obtained without perturbation, the error on the tool origin increases by 0.37mm only, and the angular error stalls. However, the pose error increases notably for the 5% perturbation case: the position error rises by 1.61mm, while the angular error is stable, even decreasing by 0.35°.

The 3D pose estimation method's sensitivity to degraded initialization was evaluated using artificially noise-contaminated ART-Net geometric primitive maps. White Gaussian noise of increasing standard deviation, namely 0.5, 1.0, 1.5, 2.0, and 2.5 for pixels values within $[0, 255]$, was added to each map computed from animal data. The pose initialization and refinement steps were performed from the 32 maps for each noise level, leading to 160 additional pose estimates. The results are reported in figure 17. They are expressed in terms of 3D euclidean distance between the estimated tool origin and tip and the ground-truth positions and angular error of the tool shaft. The standard deviation of the angular and tip position errors increases with the level of noise. It is due to some very poorly estimated poses. More specifically, 28 noisy cases gave an angular error above 20°, and 23 cases gave a tip position error above $20mm$ (which roughly corresponds to the tip length), representing 14.4% and 17.5% of the noisy dataset. A significant number of these failure cases are caused by the affine ambiguity discussed in section 4.6. The medians range from 4° to 9° and $4mm$ to $6mm$ for the angular and tip position, respectively.

Table 7: Quantitative evaluation of 3D pose estimation from animal data. The localization errors are in mm, and the orientation errors are in degrees.

Tool types	Diam. (mm)	Im. #	Orig. X	Orig. Y	Orig. Z	Tip X	Tip Y	Tip Z	Angle
Scissor	4.8	17	1.16	0.51	5.75	1.40	0.67	5.55	5.11°
Forceps	4.7	15	0.99	0.3	3.92	2.39	0.73	3.95	6.88°
Mean	4.75	32	**1.08**	**0.41**	**4.89**	**1.87**	**0.70**	**4.80**	**5.94°**
Scissor	4.8*1.01	17	1.41	0.67	5.97	1.71	0.83	5.83	5.11°
Scissor	4.8*1.05	15	1.69	0.68	7.26	1.91	0.84	7.49	4.76°

Evaluation on patient data. We evaluated the estimated 3D pose quantitatively from patient data. 10 laparoscopic images were extracted from a video of laparoscopic myomectomy from CHU Estaing, Clermont-Ferrand, France. These images were selected to show the tool touching the uterus surface. A preoperative model was built from the patient MRI



(a) (b) (c)

Figure 17: (a) Angular, (b) origin, and (c) tip position errors of the estimated tool 3D pose from noisy ART-Net geometric primitive maps. The position errors are the 3D euclidean distances between the estimated positions and the ground-truth ones. The errors are a function of an increasing standard deviation of the Gaussian white noise, namely from 0.0 to 2.5, applied on the ART-Net maps in $[0, 255]$.

and registered to the laparoscopic images using the pipeline (Collins et al., 2016). The registration result provides the ground-truth of the tool-tip depth. Specifically, the error on the estimated 3D position is expressed in that case as the shortest distance between the estimated 3D point and the registered uterus surface (figure 18 (a)). Collins et al. (2016) discards the tool as a nuisance in cases where it is visible. The reported error is hence not a residual registration error. It corresponds to the distance between the tool-tip and the registered model surface estimated independently.

The obtained mean and median value of the 3D pose error is $2.57mm$ and $1.99mm$, respectively. The median value of the error shows that 50 % of the distances lie within $1.99mm$, which is lower than the $2.52mm$ tool shaft radius. For example, in some frames in figure 18 (b, c), the tool deforms the uterus. In such cases, the estimated 3D pose is away from the registered uterus surface, which means that the reported errors overestimate the true ones.

The qualitative assessment of the 3D pose from the proposed pipeline shows that the proposed approach for 3D pose estimation of the surgical tool is robust, even for noisy, optical blurred, and motion blurred laparoscopic images. The quantitative assessment suggests a 3D position error is ranging numerically from the tool radius to the tool diameter and an angular error on the shaft axis of about 6°.

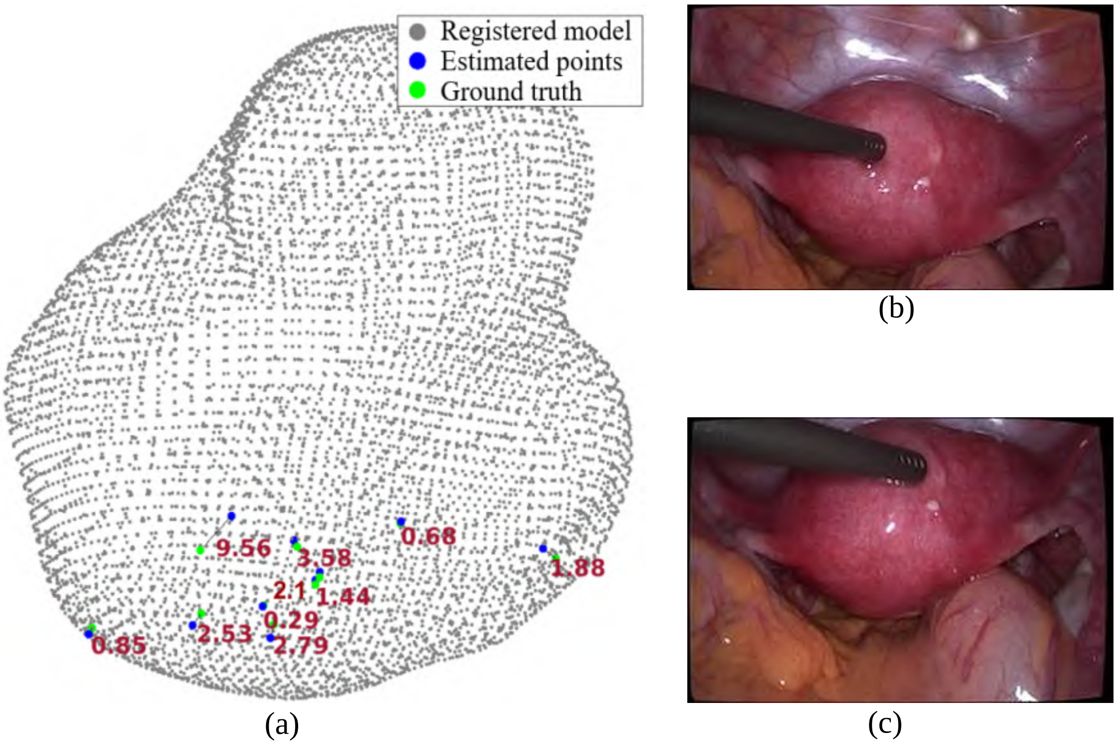


Figure 18: Quantitative evaluation of 3D poses estimation in depth from patient data. (a) A preoperative model of the uterus (vertices as grey dots) registered against 10 laparoscopic images using (Collins et al., 2016). In these images, a tool is visible and touching the uterus. Two of them are manifested in (b-c). The registration result was then used as ground-truth for pixel depth. The blue circles are the estimated 3D position of the tool-tip from the proposed framework. The green ones are their orthogonal projection onto the registered model. The distance of the estimated tool-tip position to the registered uterus model are shown in red.

4.5. Applications

We present two applications of our framework in CAL: tool-aware rendering in AR and tool-based 3D measurement.

4.5.1. Tool-aware Rendering in Augmented Reality

In AR, virtual content, named augmentation, is added to the real laparoscopy image. This content represents the internal organ anatomy. Clearly, it should not be overlaid on the tool, as this would break the surgeon's proper perception of depth. Using the segmentation mask provided by ART-Net allows us to implement this principle of tool-aware rendering, where the mask is used at the compositing stage between the rendered and the real images, denoted I_{aug} and I_{raw}, to obtain the augmented image, denoted I_{final}. In order to avoid the aliasing effect, we also smoothly blend the images near the boundary of the segmented mask. The compositing equation is:

$$I_{final} = (1-\alpha)I_{aug} + \alpha I_{raw}. \tag{4}$$

The fraction coefficient of α is taken from the median filtered tool mask. Tool-aware visualization of the augmented tool is shown in figure 19, where AR is used to visualize tumors inside the uterus. More visualizations are provided on a supplementary video with AR on the uterus on YouTube[3].

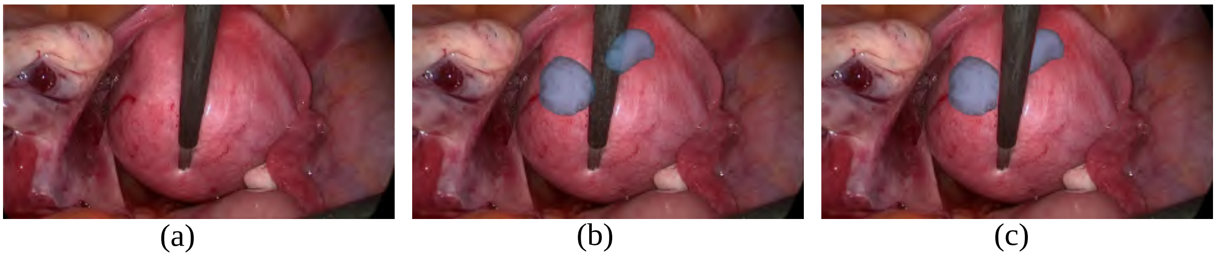


Figure 19: The proposed framework was applied to tool-aware rendering in augmented monocular laparoscopy in gynecology. (a) The input image was acquired during laparoscopic myomectomy. (b) The augmented reality system (Collins et al., 2016) overlays the image with two myomas in blue. (c) The segmentation mask from ART-Net applied to this image is used to restrict the rendering to areas unoccluded by the tool, leading to improved depth perception and realism. The entire video sequence is available on YouTube[3].

4.5.2. Anatomical Measurements

The 3D pose of the surgical tool can be used to take physical measurements of an anatomical structure when used in conjunction with a preoperative registration pipeline such as (Collins et al., 2016). We implemented this idea using two laparoscopic images of a tool in contact with a surface, intending to measure the distance between the two points on the surface. Concretely, we experimented with a sheep liver acquired *ex-vivo*, as shown in figure 20. Two points P and Q were previously marked on the liver surface, and the distance between them was measured as $38mm$ with a compliant ruler for evaluation purposes. This

[3] `https://youtu.be/Knp4JIhH3Yo`

reference distance was then compared to the distance measured between the two tool-tip positions, estimated by the proposed framework. The distance error is of $3mm$.

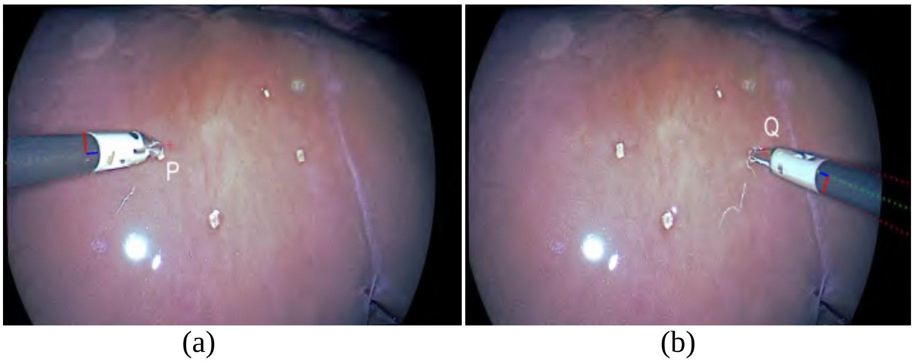


Figure 20: The proposed anatomical measurement framework tested on a static *ex-vivo* sheep liver. In these two laparoscopic images, the tool is in contact with the liver surface.

4.6. Discussions

We have proposed ART-Net, a DNN for the concurrent tool detection, segmentation, and geometric primitive extraction in laparoscopy, which was trained in an end-to-end fashion. The detection sub-network of ART-Net is based on GAP instead of the usual flattening layer. GAP has an extreme dimension reduction capability and provides useful abstract features to the model. A dropout layer used jointly with GAP increases generalisability and leads to improved model robustness. Lastly, the lightweight structure of the detection sub-network due to GAP also improves the detection rates and facilitates real-time ART-Net usage for applications such as AR guidance in CAL. The segmentation sub-network of ART-Net benefits from a new skip connection named FrG, which regains the lost spatial information by learning back the relevant features from the corresponding encoder, rather than fusing the features from the different coarseness levels of the encoder. Specifically, FrG concatenates more spatial information at the decoder's very end, providing more accurate and robust tool masks. The proposed segmentation loss combines *cross-entropy* and the IoU, providing the tools mask with smallest false-positive and false-negative rates. The geometric primitive regression sub-networks of ART-Net outperform when trained with the L_2 loss function (mean square error). Besides, FrG also boosts performance in finding the tool features and sharper gradient at the tool boundaries.

Our extensive experimental results show that the use of transfer learning to initialize the kernels in the encoder increases each sub-network's performance. *Adadelta*, as an optimizer, plays a crucial role in minimizing the loss function, where the learning rate was adapted based on a moving window of gradient updates instead of accumulating all past gradients. A comparison of the proposed integrated framework against separated frameworks was conducted. The integrated ART-Net is shown to perform substantially better than separated frameworks solving only one task amongst geometric primitive extraction and is on par for detection and segmentation. These results are reported in table 8.

Table 8: Performance comparison between the proposed SIMO ART-Net and individual sub-networks, where we use mIoU, mAL, ED, and accuracy as metrics respectively for segmentation, edge-line and mid-line, tool-tip, and detection.

Network Types	Tool Segmentation	Geometric Primitives			Tool Detection
		Edge-line	Mid-line	Tool-tip	
Individual sub-network	88.4 %	5.63°	4.15°	13.7	100.0 %
SIMO ART-Net	88.2 %	2.45°	2.23°	9.3	100.0 %

The number of training images used in our experiments is very limited, namely 508 images. It would be interesting to prepare experiments with a larger dataset in future work and measure the segmentation and geometric primitive extraction accuracy for different amounts of training images. It would allow one to quantify the number of training images from which increasing the training dataset further only marginally improves the accuracy of ART-Net.

The estimated 3D pose from the vanishing point of the obtained primitives is precise and robust. Complicated cases are, for instance, cases for which the tool pixels almost blend with the background tissue pixels near the tool boundary, where classical filtering most often fails to extract the gradient information. These cases are successfully handled by our framework, where ART-Net accurately finds the gradient information of the tool, irrespective of the tool color, shape, size, and orientation. For several images in our experiments on animal data, we have met the problem related to the affine ambiguity of 3D pose that occurs when the visible part of the tool is either far from the camera, or the tool shaft is nearly parallel to the pixel plane. In these cases, there are two solutions to the 3D pose. Resolving the ambiguity is a subject for future work. It could be achieved, for example, by using prior information about the relative position of the optical trocar, the tool trocar, and the organ or by multiple-view geometric constraints or by temporal consistency.

Integrating tool detection per tool type could also be investigated. It may lead to improved detection and segmentation results, though at the cost of losing the methods genericity. Using this information could also allow one to deal with 3D pose estimation of several tools visible simultaneously.

5. Conclusions

We have proposed an integrated approach to surgical tool detection, segmentation, and 3D pose estimation by combining statistical learning and geometry. The proposed FrG has played a crucial role in compensating for the spatial information loss due to subsampling in segmentation and regression sub-networks of ART-Net. It can also be readily used in other kinds of encoder-decoder networks for semantic segmentation. In SIMO structures, where the network comprises several sub-networks trained in an end-to-end fashion, being lightweight is one of the core requirements for real-time applications. The proposed ART-Net uses depth-wise separable convolution and GAP to reduce the number of parameters approximately 3.6 times, leading to a more general model outperforming previous work. Hence, the use of depth-wise separable convolution and GAP can improve lightweight SIMO networks. Our geometric primitive extraction approach is tool-generic, and the estimated 3D pose is precise and robust. Hence, the proposed pipeline can be applied to many other CAL applications. For instance, the estimated 3D pose could be used to resolve registration ambiguities between a preoperative organ 3D model and laparoscopic images without any additional hardware. Besides, the scale ambiguity of a 3D reconstruction obtained using Structure-from-Motion techniques could be quickly resolved by exploiting tool 3D pose.

References

Agustinos, A., Voros, S., 2015. 2d/3d real-time tracking of surgical instruments based on endoscopic image processing, in: Computer-Assisted and Robotic Endoscopy, Springer. pp. 90–100.

Al Hajj, H., Lamard, M., Conze, P.H., Cochener, B., Quellec, G., 2018. Monitoring tool usage in surgery videos using boosted convolutional and recurrent neural networks. Medical image analysis 47, 203–218.

AliceVision, 2018. Meshroom: A 3D reconstruction software. URL: `https://github.com/alicevision/meshroom`.

Allan, M., Chang, P.L., Ourselin, S., Hawkes, D.J., Sridhar, A., Kelly, J., Stoyanov, D., 2015. Image based surgical instrument pose estimation with multi-class labelling and optical flow, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer. pp. 331–338.

Allan, M., Ourselin, S., Thompson, S., Hawkes, D.J., Kelly, J., Stoyanov, D., 2012. Toward detection and localization of instruments in minimally invasive surgery. IEEE Transactions on Biomedical Engineering 60, 1050–1058.

Arel, I., Rose, D.C., Karnowski, T.P., 2010. Deep machine learning-a new frontier in artificial intelligence research [research frontier]. IEEE computational intelligence magazine 5, 13–18.

Attia, M., Hossny, M., Nahavandi, S., Asadi, H., 2017. Surgical tool segmentation using a hybrid deep cnn-rnn auto encoder-decoder, in: 2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC), IEEE. pp. 3373–3378.

Badrinarayanan, V., Kendall, A., Cipolla, R., 2017. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. IEEE transactions on pattern analysis and machine intelligence 39, 2481–2495.

Ballard, D.H., 1981. Generalizing the hough transform to detect arbitrary shapes. Pattern recognition 13, 111–122.

Bartoli, A., Lapresté, J.T., 2008. Triangulation for points on lines. Image and Vision Computing 26, 315–324.

Buell, J.F., Cherqui, D., Geller, D.A., O'Rourke, N., Iannitti, D., Dagher, I., Koffron, A.J., Thomas, M., Gayet, B., Han, H.S., Wakabayashi, G., Belli, G., Kaneko, H., Ker, C.G., Scatton, O., Laurent, A., Abdalla, E.K., Chaudhury, P., Dutson, E., Gamblin, C., D'Angelica, M., Nagorney, D., Testa, G., Labow, D., Manas, D., Poon, R.T., Nelson, H., Martin, R., Clary, B., Pinson, W.C., Martinie, J., Vauthey, J.N., Goldstein, R., Roayaie, S., Barlet, D., Espat, J., Abecassis, M., Rees, M., Fong, Y., McMasters, K., Broelsch, C., Busuttil, R., Belghiti, J., Strasberg, S., Chari, R.S., 2008. The international position on laparoscopic liver surgery: The louisville statement. Annals of surgery 250, 825–30.

Cheung, T.T., Poon, R.T., Yuen, W.K., Chok, K.S., Jenkins, C.R., Chan, S.C., Fan, S.T., Lo, C.M., 2013. Long-term survival analysis of pure laparoscopic versus open hepatectomy for hepatocellular carcinoma in patients with cirrhosis: a single-center experience. Annals of surgery 257, 506–11.

Choi, B., Jo, K., Choi, S., Choi, J., 2017. Surgical-tools detection based on convolutional neural network in laparoscopic robot-assisted surgery, in: 2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Ieee. pp. 1756–1759.

Chollet, F., 2017. Xception: Deep learning with depthwise separable convolutions, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1251–1258.

Collins, T., Chauvet, P., Debize, C., Pizarro, D., Bartoli, A., Canis, M., Bourdel, N., 2016. A system for augmented reality guided laparoscopic tumour resection with quantitative ex-vivo user evaluation, in: International Workshop on Computer-Assisted and Robotic Endoscopy, Springer. pp. 114–126.

Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L., 2009. Imagenet: A large-scale hierarchical image database, in: 2009 IEEE conference on computer vision and pattern recognition, Ieee. pp. 248–255.

Doignon, C., Graebling, P., De Mathelin, M., 2005. Real-time segmentation of surgical instruments inside the abdominal cavity using a joint hue saturation color feature. Real-Time Imaging 11, 429–442.

Feuerstein, M., Reichl, T., Vogel, J., Schneider, A., Feussner, H., Navab, N., 2007. Magneto-optic tracking of a flexible laparoscopic ultrasound transducer for laparoscope augmentation, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer. pp. 458–466.

Fuks, D., Cauchy, F., Ftriche, S., Nomi, T., Schwarz, L., Dokmak, S., Scatton, O., Fusco, G., Belghiti, J., Gayet, B., Soubrane, O., 2016. Laparoscopy decreases pulmonary complications in patients undergoing major liver resection: A propensity score analysis. Annals of surgery 263, 353–61.

Garcia-Garcia, A., Orts-Escolano, S., Oprea, S., Villena-Martinez, V., Martinez-Gonzalez, P., Garcia-Rodriguez, J., 2018. A survey on deep learning techniques for image and video semantic segmentation. Applied Soft Computing 70, 41–65.

Garcia-Peraza-Herrera, L.C., Li, W., Fidon, L., Gruijthuijsen, C., Devreker, A., Attilakos, G., Deprest, J., Vander Poorten, E., Stoyanov, D., Vercauteren, T., et al., 2017. Toolnet: holistically-nested real-time segmentation of robotic surgical tools, in: 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), IEEE. pp. 5717–5722.

Garcia-Peraza-Herrera, L.C., Li, W., Gruijthuijsen, C., Devreker, A., Attilakos, G., Deprest, J., Vander Poorten, E., Stoyanov, D., Vercauteren, T., Ourselin, S., 2016. Real-time segmentation of non-rigid surgical tools based on deep learning and tracking, in: International Workshop on Computer-Assisted and Robotic Endoscopy, Springer. pp. 84–95.

Hasan et al., 2020. Implentational details of ART-Net and datasets. `https://github.com/kamruleee51/ART-Net`.

Jaffray, B., 2005. Minimally invasive surgery. Archives of disease in childhood 90, 537–542.

Jayarathne, U.L., McLeod, A.J., Peters, T.M., Chen, E.C., 2013. Robust intraoperative us probe tracking using a monocular endoscopic camera, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer. pp. 363–370.

Jin, A., Yeung, S., Jopling, J., Krause, J., Azagury, D., Milstein, A., Fei-Fei, L., 2018. Tool detection and operative skill assessment in surgical videos using region-based convolutional neural networks, in: 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), IEEE. pp. 691–699.

Jin, Y., Li, H., Dou, Q., Chen, H., Qin, J., Fu, C.W., Heng, P.A., 2020. Multi-task recurrent convolutional network with correlation loss for surgical video analysis. Medical image analysis 59, 101572.

Kaiser, L., Gomez, A.N., Shazeer, N., Vaswani, A., Parmar, N., Jones, L., Uszkoreit, J., 2017. One model to learn them all. arXiv:1706.05137 .

Krizhevsky, A., Sutskever, I., Hinton, G.E., 2012. Imagenet classification with deep convolutional neural networks, in: Advances in neural information processing systems, pp. 1097–1105.

Krupa, A., Gangloff, J., Doignon, C., De Mathelin, M.F., Morel, G., Leroy, J., Soler, L., Marescaux, J., 2003. Autonomous 3-d positioning of surgical instruments in robotized laparoscopic surgery using visual servoing. IEEE transactions on robotics and automation 19, 842–853.

Kurmann, T., Neila, P.M., Du, X., Fua, P., Stoyanov, D., Wolf, S., Sznitman, R., 2017. Simultaneous recognition and pose estimation of instruments in minimally invasive surgery, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer. pp. 505–513.

Laina, I., Rieke, N., Rupprecht, C., Vizcaíno, J.P., Eslami, A., Tombari, F., Navab, N., 2017. Concurrent segmentation and localization for tracking of surgical instruments, in: International conference on medical image computing and computer-assisted intervention, Springer. pp. 664–672.

Lepetit, V., Moreno-Noguer, F., Fua, P., 2009. Epnp: An accurate o(n) solution to the pnp problem. International Journal Of Computer Vision 81, 155–166.

Lin, M., Chen, Q., Yan, S., 2013. Network in network. arXiv:1312.4400 .

Long, J., Shelhamer, E., Darrell, T., 2015. Fully convolutional networks for semantic segmentation, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 3431–3440.

Melo, R., Barreto, J.P., Falcao, G., 2011. A new solution for camera calibration and real-time image distortion correction in medical endoscopy–initial technical evaluation. IEEE Transactions on Biomedical Engineering 59, 634–644.

Milletari, F., Rieke, N., Baust, M., Esposito, M., Navab, N., 2018. Cfcm: Segmentation via coarse to fine context memory, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer. pp. 667–674.

Moré, J.J., 1978. The levenberg-marquardt algorithm: implementation and theory, in: Numerical analysis. Springer, pp. 105–116.

Mottaghi, R., Chen, X., Liu, X., Cho, N.G., Lee, S.W., Fidler, S., Urtasun, R., Yuille, A., 2014. The role of context for object detection and semantic segmentation in the wild, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 891–898.

Nwoye, C.I., Mutter, D., Marescaux, J., Padoy, N., 2019. Weakly supervised convolutional lstm approach for tool tracking in laparoscopic videos. International journal of computer assisted radiology and surgery 14, 1059–1067.

Odena, A., Dumoulin, V., Olah, C., 2016. Deconvolution and checkerboard artifacts. Distill 1, e3.

Pakhomov, D., Premachandran, V., Allan, M., Azizian, M., Navab, N., 2019. Deep residual learning for instrument segmentation in robotic surgery, in: International Workshop on Machine Learning in Medical Imaging, Springer. pp. 566–573.

Pratt, P., Jaeger, A., Hughes-Hallett, A., Mayer, E., Vale, J., Darzi, A., Peters, T., Yang, G.Z., 2015. Robust ultrasound probe tracking: initial clinical experiences during robot-assisted partial nephrectomy. International journal of computer assisted radiology and surgery 10, 1905–1913.

Rasmus, A., Berglund, M., Honkala, M., Valpola, H., Raiko, T., 2015. Semi-supervised learning with ladder networks, in: Advances in neural information processing systems, pp. 3546–3554.

Ronneberger, O., Fischer, P., Brox, T., 2015. U-Net: Convolutional networks for biomedical image segmentation, in: International Conference on Medical image computing and computer-assisted intervention, Springer. pp. 234–241.

Salah, Z., Preim, B., Elolf, E., Franke, J., Rose, G., 2011. Improved navigated spine surgery utilizing augmented reality visualization, in: Bildverarbeitung für die Medizin 2011. Springer, pp. 319–323.

Schneider, C.A., Rasband, W.S., Eliceiri, K.W., 2012. Nih image to imagej: 25 years of image analysis. Nature methods 9, 671–675.

Simonyan, K., Zisserman, A., 2014. Very deep convolutional networks for large-scale image recognition. arXiv:1409.1556 .

Smith, L.N., 2017. Cyclical learning rates for training neural networks, in: 2017 IEEE Winter Conference on Applications of Computer Vision (WACV), IEEE. pp. 464–472.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R., 2014. Dropout: a simple way to prevent neural networks from overfitting. The journal of machine learning research 15, 1929–1958.

Suzuki, S., et al., 1985. Topological structural analysis of digitized binary images by border following. Computer vision, graphics, and image processing 30, 32–46.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A., 2015. Going deeper with convolutions, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1–9.

Twinanda, A.P., Shehata, S., Mutter, D., Marescaux, J., De Mathelin, M., Padoy, N., 2016. Endonet: a deep architecture for recognition tasks on laparoscopic videos. IEEE transactions on medical imaging 36, 86–97.

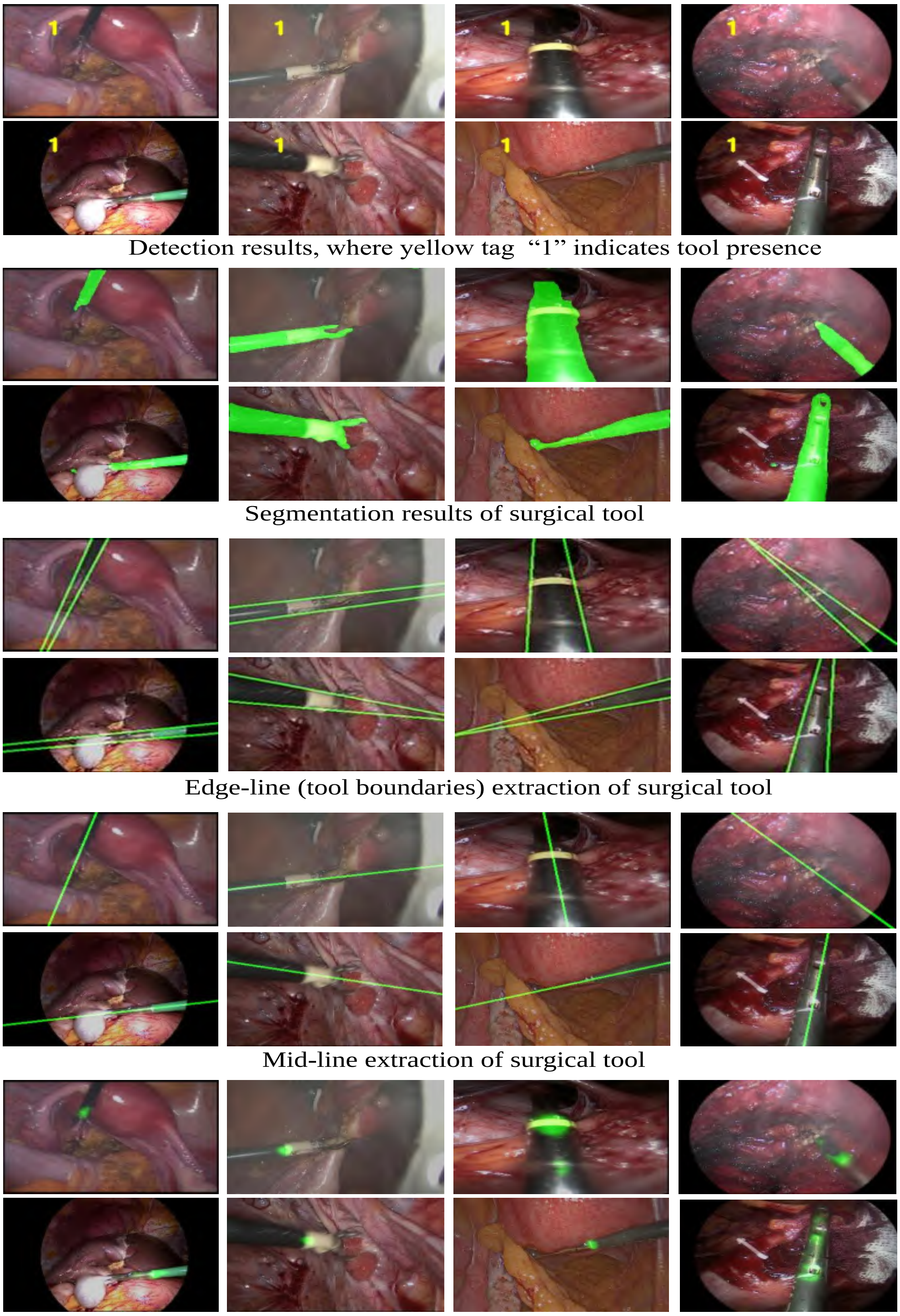
Wang, M.L., Wu, J.J., Lee, P.Y., Hu, M.H., Kumar, A., Chen, L.X., Liu, K.C., Marescaux, J., Nicolau, S., Vemuri, A., et al., 2013. A landmark based registration technique for minimally invasive spinal surgery, in: 2013 IEEE International Symposium on Consumer Electronics (ISCE), IEEE. pp. 235–236.

Wang, S., Raju, A., Huang, J., 2017. Deep learning based multi-label classification for surgical tool presence detection in laparoscopic videos, in: 2017 IEEE 14th International Symposium on Biomedical Imaging (ISBI 2017), IEEE. pp. 620–623.

Wei, G.Q., Arbter, K., Hirzinger, G., 1997. Automatic tracking of laparoscopic instruments by color coding, in: CVRMed-MRCAS'97, Springer. pp. 357–366.

Zeiler, M.D., 2012. Adadelta: an adaptive learning rate method. arXiv:1212.5701 .

Appendix A. Additional Results



Detection results, where yellow tag "1" indicates tool presence

Segmentation results of surgical tool

Edge-line (tool boundaries) extraction of surgical tool

Mid-line extraction of surgical tool

Tool-tip extraction of surgical tool

Figure A.21: Results delivered by our framework under challenging conditions.

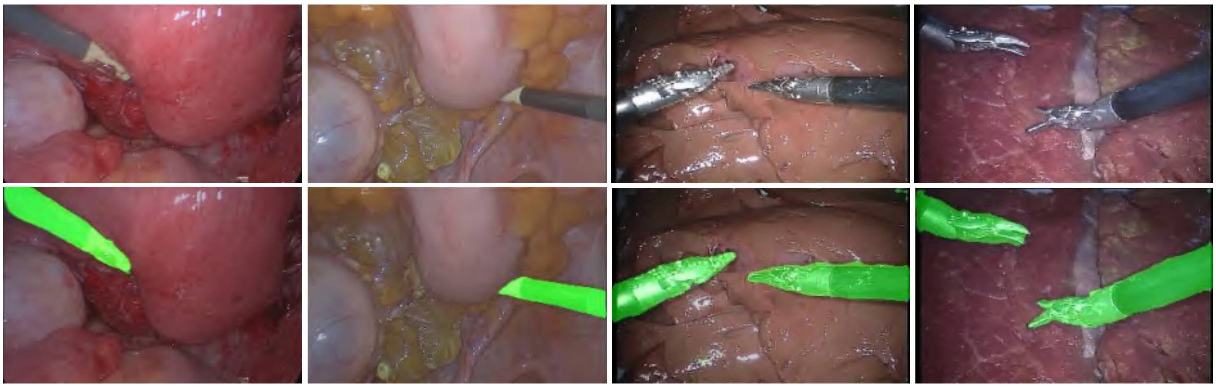


Figure A.22: Additional tool segmentation results with the presence of multiple tools (right) and with tool occlusions (left). First row: input images; second row: segmentation results from ART-Net. Although not trained to segment multiple tools visible in the same image, ART-Net is shown to perform well in that case, as well as to be robust to occlusions.