Detection and Localization of Robotic Tools in Robot-Assisted Surgery Videos Using Deep Neural Networks for Region Proposal and Detection

Duygu Sarikaya,* Jason J. Corso, and Khurshid A. Guru

***Abstract*— Video understanding of robot-assisted surgery (RAS) videos is an active research area. Modeling the gestures and skill level of surgeons presents an interesting problem. The insights drawn may be applied in effective skill acquisition, objective skill assessment, real-time feedback, and human–robot collaborative surgeries. We propose a solution to the tool detection and localization open problem in RAS video understanding, using a strictly computer vision approach and the recent advances of deep learning. We propose an architecture using multimodal convolutional neural networks for fast detection and localization of tools in RAS videos. To the best of our knowledge, this approach will be the first to incorporate deep neural networks for tool detection and localization in RAS videos. Our architecture applies a region proposal network (RPN) and a multimodal two stream convolutional network for object detection to jointly predict objectness and localization on a fusion of image and temporal motion cues. Our results with an average precision of 91% and a mean computation time of 0.1 s per test frame detection indicate that our study is superior to conventionally used methods for medical imaging while also emphasizing the benefits of using RPN for precision and efficiency. We also introduce a new data set, ATLAS Dione, for RAS video understanding. Our data set provides video data of ten surgeons from Roswell Park Cancer Institute, Buffalo, NY, USA, performing six different surgical tasks on the daVinci Surgical System (dVSS) with annotations of robotic tools per frame.**

***Index Terms*— Object detection, multi-layer neural network, image classification, laparoscopes, telerobotics.**

I. Introduction

ROBOT-ASSISTED surgery (RAS) is the latest form of development in today's minimally invasive surgical technology. The robotic tools help the surgeons complete complex motion tasks during procedures with ease by translating the surgeons' real-time hand movements and force on the tissue into small scale ones. Despite its advances in minimally invasive surgery, the steep learning curve of the robot-assisted surgery devices remains a disadvantage [1]. Translation of the surgeons' movements via the robotic device is challenged by a loss of haptic sensation. Surgeons usually feel comfortable with the procedures only after they have completed procedures on 12-18 patients [1]. The traditional mode of surgical training (apprenticeship) fails to answer the needs of today's RAS training. The conventional approach relies heavily on observational learning and operative practice [2]. Moreover, the evaluation of the training also relies on the subjective observance of an experienced surgeon, ideally measured by a senior surgeon with a scoring system [3]. The need for universally accepted and validated metrics and quantitative skill assessment via automation is addressed in the community [2]. Early identification of technical competence in surgical skills is expected to help tailor training to personalized needs of surgeons in training [2], [3]. Moreover, we believe that automated feedback or human-robot collaborative surgeries could greatly benefit novice surgeons and their patients. As such, we have investigated the development of such systems with video understanding via computer vision using the video data recorded during surgical tasks on the daVinci Surgical System (dVSS®). Figure 2 shows sample frames from recorded video data of surgeons performing on training sets.

We approach the problem of video understanding of RAS videos as modeling the motions of surgeons. Modeling gestures and skills depends highly on the motion information and the precise trajectories of the motion. We argue that, detecting the tools and capturing their motion in a precise manner is an important step in RAS video understanding. The first step for tracking the tools in the video is to robustly detect the presence of a tool and then localize it in the image. Tool detection and localization is hence the focus of our study.

Advances in object detection had plateaued until the recent reintroduction of deep neural networks into the computer vision community with large-scale data object classification tasks. The deep neural network trained via back-propagation through layers of convolutional filters by LeCun *et al.* [4], [5] had an outstanding performance on large amounts of training data. On large-scale recognition challenges like ImageNet [6],

Manuscript received January 6, 2017; accepted January 30, 2017. Date of publication February 8, 2017; date of current version June 29, 2017. *Asterisk indicates corresponding author.*

*D. Sarikaya is with the Department of Computer Science and Engineering, SUNY Buffalo, NY 14260-1660 USA (e-mail: duygusar@buffalo.edu).

J. J. Corso is with the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109 USA.

K. A. Guru is with the Applied Technology Laboratory for Advanced Surgery, Roswell Park Cancer Institute, Buffalo, NY 14263 USA.

Digital Object Identifier 10.1109/TMI.2017.2665671

Fig. 1. We propose an end-to-end deep learning approach for tool detection and localization in RAS videos. Our architecture has two separate CNN processing streams on two modalities: the RGB video frame and the RGB representation of the optical flow information of the same frame. We convolve the two separate input modalities and get their convolutional feature maps. Using the RGB image convolutional features, we train a Region Proposal Network (RPN) to generate object proposals. We use the region proposals and the feature maps as input to our object classifier network. The last layer features of these streams are later fused together before classifier.

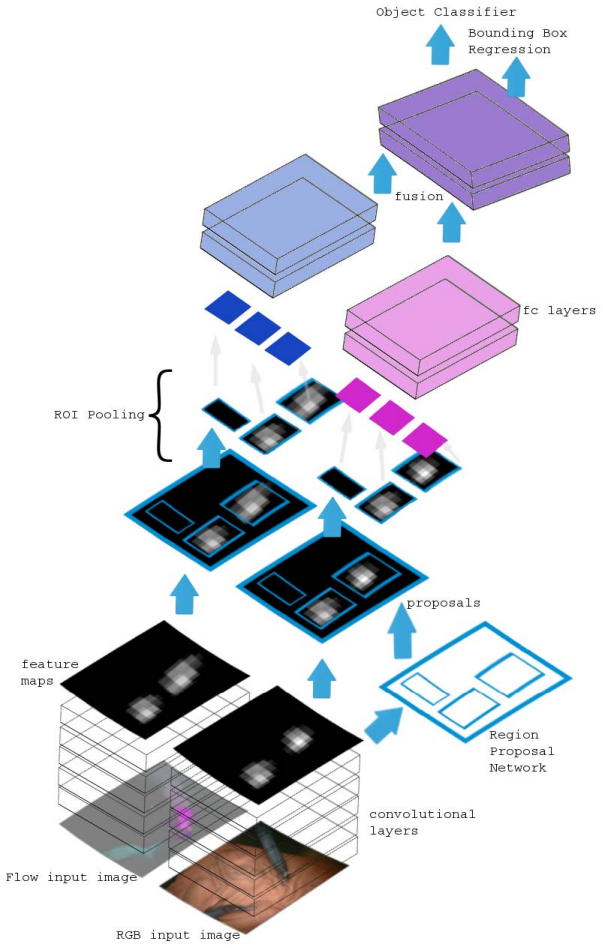


Fig. 2. Sample frames of the video data recorded during surgical training tasks on the daVinci Surgical System (dVSS®).

similar approaches have now proven that deep neural networks could be used with object recognition and classification tasks [6], [7]. However, in the medical field these advances are yet unused, to the best of our knowledge; so, we propose a novel approach of using deep convolutional neural networks (CNN) for fast detection and localization for video understanding of RAS videos. No study we are aware of has addressed the use of CNN in tool detection and localization in RAS video understanding and our study will be the first.

In this paper, we address the problem of object detection and localization, specifically the surgical robotic tool, in RAS videos as a deep learning problem. We propose complete solutions for detection and localization using CNNs. We apply Region Proposal Networks (RPN) [8] jointly with a multimodal object detection network for localization; as such, we simultaneously predict object region proposals and objectness scores. Figure 1 shows an overview of our system. Our results indicate that our study is superior to conventionally used methods for medical imaging with an Average Precision (AP) of 91% and a mean computation time of 0.1 seconds per test frame.

II. Related Work

Many early approaches to detecting and tracking robotic tools in surgery videos use markers and landmarks to reduce the problem to a simple computer vision problem of color segmentation or thresholding [9], [10]. Using color markers or color coding the tools are examples of this approach. Another example of a marker is a laser-pointing instrument holder used to project laser spots on the scene [11]. The works of Zhang and Payandeh [12] and Reiter *et al.* [13] introduce a barcode marker. However, these methods require additional manufacturing and raise concerns on biocompatibility. Moreover, having to use additional instruments in minimally invasive surgical settings could be challenging [14].

More recent approaches focus on tracking the tool via a per video initialization and/or using additional modalities such as kinematic data. Du *et al.* [15], Lowe [16], and Ballard [17] develop a 2D tracker based on a Generalized Hough Transform using SIFT features and use this to initialize a 3D tracker at each frame to recover instrument pose. This method assumes that they have the 3D pose of the instrument in the first frame and they use this information to initialize a 2D bounding box which is then used as a reference point for tracking. Allan *et al.* uses Random Forests to fuse region based constraints based on multi-label probabilistic region classifications with low level optical flow information [18]. Instead of an offline learning approach, they train their Random Forest using a manual segmentation of a single frame with a tool positioned in front of the scene. This approach poses a drawback as it does not allow their system to operate on different surgical setups without re-training. Capturing kinematic data of the robotic

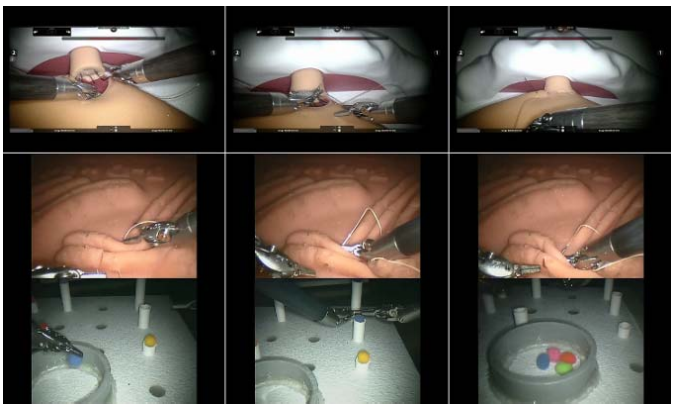


Fig. 3. The tools in RAS videos vary in pose, articulation and parts of the tool might be occluded or missing in the frame.

console as an additional feature to help with tool detection and localization could be an alternative, however, it requires additional instruments, recording, and preprocessing in order to be used as motion information. Recent approaches such as the work presented in Reiter *et al.*'s [19], creates templates using different kinematic configurations of the virtual renderings of a CAD model of the tool. They further refine their configuration estimation by matching gradient orientation templates to the real image.

There are strictly vision based approaches as well. In their work, Sznitman *et al.* [20] learn a multiclass classifier based on tool-parts detectors. During training they use image windows and evaluate the image features that compute the proportions of edges in different locations and orientations in relation to that window. At test time, to localize the tool, they evaluate the classifier in a sliding window fashion; at each image location and at multiple image scales. Although they use an early stopping algorithm to reduce the time, we believe that it is a priority to offer a solution that is more effective and efficient. One of the common computer vision approaches is using shape matching. The challenge of detection by fitting shape models is that the objects in the RAS videos in study show a high-degree of variation in shape, pose, and articulation. The parts of the objects might even be occluded or missing in the frame as seen in Figure 3. So the approaches that follow a rigid appearance and global shape model are usually not sufficient. In contrast, the Deformable Parts Model (DPM) by Felzenszwalb *et al.* [21], [22] consists of a star-structured pictorial model linking the root of an object to its parts using deformable springs. This model has proven to be successful in object detection as it captures the occlusions and articulations. One of the recent, surgical tool detection applications of DPM [21], [22] is the product of experts tool detector by Kumar *et al.* [14]. We compare our results to DPM [21], [22] as it is often successfully used in recent successful applications in similar domains.

In this paper, we propose a novel end-to-end approach of using deep convolutional neural networks (CNN) for fast tool detection and localization for video understanding of RAS videos. We apply Region Proposal Networks (RPN) [8] jointly with a multimodal object detection network for localization. Our architecture, based on the work of Zeiler and Fergus [24] and developed to support multimodality, starts with two separate CNN processing streams on two modalities: the RGB video frame and the RGB representation of the optical flow information of the same frame as temporal motion cues. RPN is a deep, fully convolutional network that outputs a set of region proposals based on the convolutional feature maps of the RGB image inputs. We adopt Fast R-CNN [25] for the object detection task, we use the region proposal boxes of RPN with the convolutional features as input for the detection network streams on both modalities. The last layer features of these streams are fused together.

Our study differs from the former approaches as we focus on proposing a solution using strictly computer vision instead of using additional modalities that require additional equipment to capture; such as kinematic data. With our study, we propose to make use of recent advances of Convolutional Neural Networks with the hope that it will serve as a benchmark for tool tracking to move towards deep learning. We also prioritize an efficient and effective way of localizing the tools in the images, our study demonstrates the invaluable contribution of RPNs to the problem of tool detection and localization in RAS videos. With our new architecture that incorporates two modalities, we make use of not only the visual object features but also the temporal motion cues. This approach helps us improve the detection of the tools by decreasing false positives. Our experiments show that our method is able to detect and localize the tools fast with superior accuracy. Our proposed method does not require initialization, so it can be used on new video data without re-training. It could be used to automatically initialize tracking algorithms in RAS videos.

III. Dataset

RAS video understanding may be particularly challenging as the view of the operating site is limited and recorded via endoscopic cameras. Tracking the free movement of the surgeons in unconstrained scenarios requires camera movement and zoom. The tools in RAS videos vary in pose, articulation and their parts might be occluded or missing in the frame (refer to Figure 3 for visual examples). There are frequently other objects in use and in motion such as the needle, suture, clamps or other objects used in training. The tissue that the surgeon operates on might move or deform, show variation in shape and occlude the tool.

We have noticed that the RAS datasets for public use are limited. The only comprehensive RAS datasets we know of that is open for public use are JIGSAWS by Gao *et al.* [26] and the very recently released m2cai16-workflow and m2cai16-tool datasets [27]. Neither of these datasets provide tool annotations. JIGSAWS is quite restricted as it does not include artifacts, camera movement and zoom, or a wide range of free movement. For this reason, we have built a new, more challenging dataset; ATLAS Dione, that leverages the data gathered for earlier works of Guru *et al.* [3] and Stegemann *et al.* [28]. We have used the video data and prepared manual annotations for RAS video understanding problems. This new video dataset is a core contribution of our work and we will release it for tool detection and localization

TABLE I
PROPERTIES OF THE DATASET

Main Dataset Skill Level	*Task*	*Subtask*	*Number of Videos*	*Number of Frames*
Basic Skills	Ball Placement Task	Using 1 Arm	7 videos	
		Using 2 Arms	7 videos	
	The Ring Peg Transfer Task	Using 1 Arm	7 videos	
		Using 2 Arms	7 videos	
	Suture Pass Task	Put Through	7 videos	
		Pull Through	7 videos	
Intermediate Skills	Suture and Knot Tie Task	Suture Pick Up	7 videos	
		Suture Pull Through	7 videos	
		Suture Tie	7 videos	
Advanced Skills	Urethrovesical Anastomosis(UVA)	UVA Pick Up	7 videos	
		UVA Pull Through	7 videos	
		UVA Tie	7 videos	
Total			***84 videos***	***18782 frames***
Additional Test Set of compiled subtasks			15 videos	*3685 frames*
Total			***99 videos***	***22467 frames***

purposes upon publication. Despite being a phantom setting, our dataset is challenging as it has camera movement and zoom, free movement of surgeons, a wider range of expertise levels, background objects with high deformation, and annotations include tools with occlusion, change in pose and articulation or when they are only partially visible in the scene.

ATLAS Dione provides video data of ten subjects performing six different surgical tasks on the dVSS®. The ten surgeons who participated in this IRB-approved study (I 228012) work at Roswell Park Cancer Institute (RPCI) (Buffalo, NY). The difficulty and complexity of the tasks vary. These tasks include basic RAS skill tasks which are part of the Fundamental Skills of Robotic Surgery (FSRS) curriculum [28] and also the procedure-specific skills required for the Robotic Anastomosis Competency Evaluation (RACE) [3]. The subjects, surgeons from RPCI, are classified by different expertise levels of beginner (BG), combined competent and proficient (CPG), and expert (EG) groups based on the Dreyfus model [31]. For tool detection and localization purposes, we manually label both the left and right tools in use by providing exact bounding boxes of tool locations. In addition to these, following our future works, expertise levels of the subjects, task videos with the beginning and ending timestamps for each subtask will also be released for surgical activity recognition research.

The robotic skill groups based on the difficulty and complexity of the tasks are as follows (Refer to Figure 4 for sample frames of these skill groups):

1) Basic Skills: Ball Placement Task, Suture Pass Task and Ring Peg Transfer Task.
2) Intermediate Skills: Placement of Simple Suture with Knot Tying.
3) Advanced Skills: Performance of Urethrovesical Anastomosis (UVA) on an Inanimate Model.

Please refer to Table I to see properties of the dataset.

A. Tool Annotation

For the tool detection and localization, we annotate bounding boxes for both left and right tools seen in the videos. Bounding boxes are manually annotated with the guidance of an expert RAS surgeon. For efficient annotation, we have customized the tools provided by Caltech Pedestrian Detection Benchmark by Dollár *et al.* [29] to our problem and used it to annotate each frame in the video clips. We provide each frame (each of size 854×480 pixels) of the RAS videos in JPEG format. The annotations are provided in XML templates in VOC format [30].

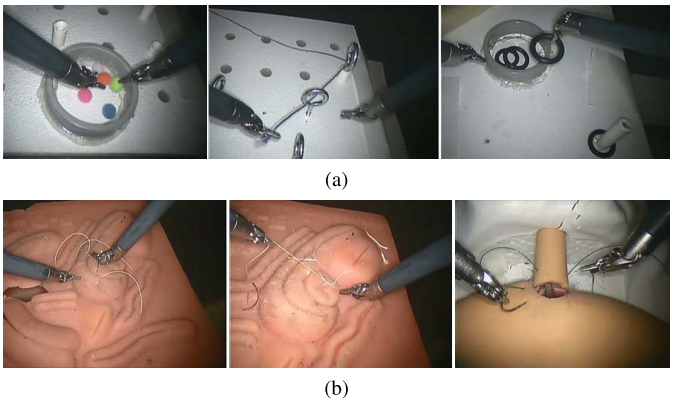


Fig. 4. Sample frames of tasks under different robotic skill groups based on the difficulty and complexity. (a) Basic Skills: Ball Placement Task, Suture Pass Task and Ring Peg Transfer Task. (b) Intermediate Skills: Placement of Simple Suture with Knot Tying. Advanced Skills: Performance of Urethrovesical Anastomosis (UVA) on an Inanimate Model.

B. Subject Demographics

We recorded 10 surgeons from RPCI performing the given tasks on (dVSS®). Out of these 10 subjects, 2 of them are residents, 3 are fellows and 5 are practicing robotic surgeons. While 3 of them have more than 10 years of experience, 2 of them have between 2 to 5 years of experience and the remaining 5 subjects are still in training. Two of our subjects have performed over 500 robot-assisted procedures. The ten subjects are assigned to beginner (BG), combined competent and proficient (CPG), and expert (EG) groups based on the Dreyfus model [31].

Upon publication of this study, the dataset with tool annotations will be available for download to encourage

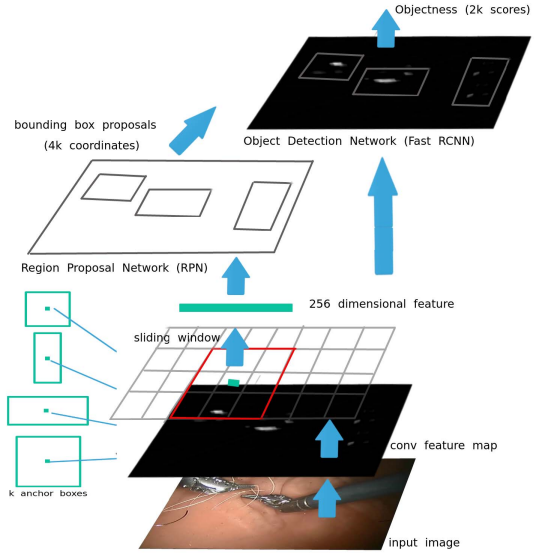


Fig. 5. We use a Region Proposal Network that proposes regions, which are used by a detector network. We slide a small network over the convolutional feature map. Each spatial window over the feature map is mapped to a 256 dimensional feature which is fed into the system. RPN outputs object region proposals, each with an objectness score on whether the region is of a tool or not. (Please refer to IV for details of our Method.)

further research in RAS video understanding. (Further information will be provided at www.roswellpark.edu/education/atlas-program.)

IV. Method

We propose an end-to-end deep learning approach for tool detection and localization in RAS videos. Our architecture, based on the work of Zeiler and Fergus [24], has two separate CNN processing streams on two modalities: the RGB video frame and the RGB representation of the optical flow [32] information of the same frame. The last layer features of these streams are later fused together. We first convolve the two separate input modalities and get their convolutional feature maps. Using the RGB image convolutional features, we train a Region Proposal Network (RPN) [8] to generate object proposals. Figure 1 shows an overview of our system.

Figure 5 provides a visual overview of the region proposal network. RPN uses the convolutional feature maps of the RGB input for generating region proposals. Each of these proposals have an objectness score. In our study, we use an architecture based on the one proposed by Zeiler and Fergus [24] with five convolutional layers followed by fully connected layers. As proposed by the work of Ren *et al.* [8], we slide a network over the convolutional feature map of the $conv_5$ layer in a sliding-window fashion. This network is fully connected to a spatial window of the convolutional feature map with a 3×3 convolutional layer. Then each 3×3 window is mapped to a lower-dimensional; fixed 256 dimensional feature vector. This feature vector is then used as input for a box regression layer and a box classification layer. The regression layer outputs $4k$ values: the bounding box coordinates for each of the k region proposals. The classification layer outputs $2k$ scores that estimate probability of each of the k regions being of object class or not. Region proposals are relative reference boxes to anchors centered at each sliding window. Each anchor is related with a scale of size 128, 256, and 512 pixels and aspect ratios of $1:1$, $1:2$, and $2:1$ resulting in 9 anchors at each sliding window and WHk translation invariant anchors in total where the convolutional feature map size is $W \times H$. At the training step, each anchor is given a binary class label according to Intersection-over-Union (IoU) overlap with a ground-truth box. If the IoU is higher than 0.7, the anchor is given a positive object class label whereas if the IoU is smaller than 0.3, the anchor is given a negative label. The remaining anchors are considered neutral and are not used for training purposes.

We use these region proposal boxes from the RPN and the convolutional features of both modalities, as input to the ROI pooling layer, as introduced by Girschick [25], for each stream. After the last fully connected layers, the features of both streams are concatenated and fused together in a fully connected layer before the loss layers. We train region proposal network and the multimodal object detection network jointly, with reference to the approximate joint training; that is, we ignore the derivate with respect to the proposal boxes coordinates as explained by Ren *et al.* [8]. We also initiate our region proposal network with the RGB modal image only, while we convolve the two modality pair of images and then fuse their high level features for the detection network. Our experiments show that this approach is able to produce competitive results with a modest training time.

A. Loss Function for Learning

We use a multitask objective function [33], [34] to enable learning parameters over our full network concurrently.

In our work, as proposed in Faster R-CNN [8], we minimize an objective function following the multi-task loss:

$$L(\{p_i\},\{t_i\}) = \frac{1}{N_{cls}} \sum_i L_{cls}(p_i, p_i^*) + \lambda \frac{1}{N_{reg}} \sum_i p_i^* L_{reg}(t_i, t_i^*) \quad (1)$$

Recall that the regression layer outputs $4k$ values; the bounding box coordinates for each of the k region proposals while the classification layer outputs $2k$ scores that estimate probability of each of the k region proposals being of object class or not. The classification layer outputs a discrete probability $\{p_i\}$ $p = (p_0, \ldots, p_K)$, over $K + 1(background/non-object)$ categories and the regression layer outputs $\{t_i\}$ bounding-box regression offsets; a predicted tuple $t^u = (t_x^u, t_y^u, t_w^u, t_h^u)$ for class u. In Equation (1), i is the index of an anchor and p_i is the predicted probability of anchor i being an object. According to the IoU overlap, p_i^*, the ground truth label, is 1 if the anchor is positive, and is 0 if the anchor is negative. t_i is the offset of the predicted bounding box where t_i^* is the offset of the ground-truth box associated with a positive anchor. The classification loss L_{cls} is log loss over classes for whether it is an object or not:

$$L_{cls}(p, u) = -\log p_u \quad (2)$$

for true class u.

For the regression loss (L_{reg}), we use $L_{reg}(t_i, t_i^*) = R(t_i - t_i^*)$, where R is the robust loss (smooth L1) function shown below:

$$smooth_{L1}(x) = \begin{cases} 0.5x^2 & \text{if } |x| < 1 \\ |x| - 0.5 & \text{otherwise} \end{cases} \quad (3)$$

Regression loss is activated only for positive anchors and does not contribute otherwise. The two terms are normalized with N_{cls} and N_{reg} and a balancing weight λ, set to 10, that controls the balance between the two task losses. cls is normalized by the mini-batch size (i.e., N_{cls} = 256) and the reg is normalized by the number of anchor locations. For a convolutional feature map of a size $W \times H$, there are WHk anchors in total.

We use the bounding-box regression to predict more precise boundaries and to improve localization. At each object proposal bounding box, predictions are refined with the help of the anchors and regression. The features are of a fixed size. However, a set of k bounding-box regressors for each proposed box, called anchors, are learned. Each of these anchors are responsible for a scale and an aspect ratio (Figure 5).

A transformation that maps an anchor of a proposed box P to a nearby ground-truth box G is learned using the regression suggested by Girschick *et al.* [35]. $P^k = (P_x^i, P_y^i, P_w^i, P_h^i)$ is defined to be the pixel coordinates of the center of an anchor box of a proposal P with P's width and height. Similarly, the ground truth pair is $G = (G_x, G_y, G_w, G_h)$. The transformation between the pairs of P and the nearest ground-truth box G is parameterized as a scale-invariant translation of the center of P's bounding box $d_x(P), d_y(P)$ and log-space translations of the width and height of P's bounding box.

We apply the transformations below from P into G:

$$\begin{aligned} \hat{G}_x &= P_w d_x(P) + P_x \\ \hat{G}_y &= P_h d_y(P) + P_y \\ \hat{G}_w &= P_w \exp(d_w(P)) \\ \hat{G}_h &= P_h \exp(d_h(P)) \end{aligned}$$

Each function $d_\star(P)$ is a linear function of the pool5 features of P. Assuming $d_\star(P) = w^T \Phi_5(P)$

We learn $w_\star$ by ridge regression:

The regression targets $t_\star$ for the training pair (P, G) are then defined as:

$$\begin{aligned} t_x &= (G_x P_x)/P_w \\ t_y &= (G_y P_y)/P_h \\ t_w &= \log(G_w/P_w) \\ t_h &= \log(G_h/P_h) \end{aligned}$$

which we solve as a standard regularized least squares problem.

B. Optimization

The RPN is a fully convolutional network trained end-to-end with back-propagation and stochastic gradient descent (SGD). To train this network efficiently, we first sample N images and then sample R/N anchors from each image pair. We also randomly sample a top-ranking fixed number of anchors in each modality image to compute the loss function of a mini-batch and keep a ratio of 1 : 1 of positive and negative anchors.

During the optimization we initialize the new layers by weights from a zero-mean Gaussian distribution with standard deviation 0.01, while using the pre-trained ImageNet [6] model to initialize the rest. Transferring weights from pre-trained Imagenet model has helped greatly with initialization of RPN networks and also with the RGB image convolutional features. It has also shown great benefits by addressing the problem of overfitting and has decreased the fluctuations while our model converged. Unfortunately, we couldn't find a suitable dataset model to train the newly introduced layers for the optical flow modality, we have initialized these layers by weights from a zero-mean Gaussian distribution. This has led to some fluctuations while the model converged, however, in the end, has shown improvement with accuracy.

We set the learning rate to 0.001, with momentum of 0.9 and a weight decay of 0.0005. We set the number of iterations to 70k, we have decided on this number to be optimal after experimenting with a range of 30k to 120k iterations. 70k has shown comparable results to 120k while taking much less time for training.

C. Detection at Test Time

During testing we apply the fully convolutional RPN to the entire image. To reduce the redundancy of overlapping proposals, we use non-maximum suppression (NMS) with the threshold of 0.7 on the proposals based on their objectness scores. Then we use the top ranking fixed number of proposals for detection at test.

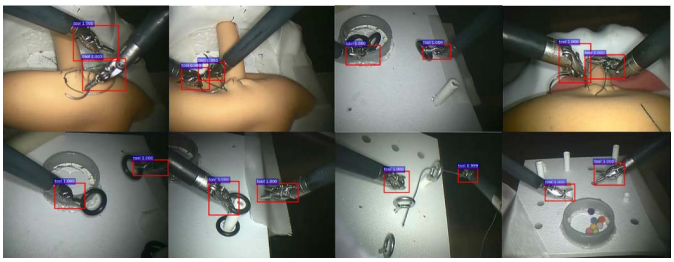
V. Experiments and Evaluation

We evaluate our architecture on the ATLAS Dione dataset using all the 99 videos for either training or testing. In order to experiment how stable the average precision results are, we do a ten fold experiment. We split 90 of the videos for training and the rest 9 for testing for each experiment setup. The videos are randomized for each experiment and we never use the frames extracted from the same video for both training and testing for the reason they might be too similar. Our dataset has only one class object; that is, the robotic tool and the background class. We use a setting of multiple 2.62GHz CPU processors and Geforce GTX1080 with computation capability of 6.1, which have allowed us to run our experiments faster with less memory consumption with cuDNN library. We use PASCAL VOC evaluation [30] to evaluate the accuracy of our detections.

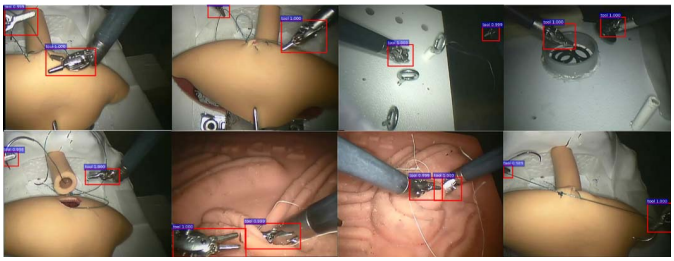
Our experimental results reach an Average Precision (AP) of 91% (90.65%). It takes 7.22 hours to train a model with 70k iterations and a mean computation time of 0.103 seconds to detect the tools in each test frame, given that we are working with a set of already computed optical flow images. We compare our experimental results with the original architecture proposed by Ren *et al.* [8], Fast R-CNN using EdgeBoxes proposals [25], [36] and Deformable Parts Model (DPM) suggested by Felzenszwalb *et al.* [21], [22], which is a proven state of the art method in medical domain and tool detection in

TABLE II
EXPERIMENTS AND EVALUATION

Method	*Mean Average Precision*	*Detection Time (per frame)*
RPN+Fast R-CNN Detection, Multimodal	91% (90.65%)	0.103 seconds + optical flow computation (a few seconds [40])
RPN+Fast R-CNN Detection (Faster R-CNN)	90% (90.39%)	0.059 seconds
Edge Boxes+Fast R-CNN Detection (Fast R-CNN)	20%	0.134 seconds for detection + 2 seconds for region proposal
Deformable Parts Model (DPM)	76% (83% with bounding box regression)	2.3 seconds



(a)



(b)

Fig. 6. Sample detection results. (a) Sample detection results of regular scenes with their scores. (b) Sample detection results of challenging scenes with their scores. Here we can see confident detections with precise boundaries even when the tool is occluded or partially out of screen.

RAS videos [14]. The architecture proposed by Ren *et al.* [8] scores 90% (90.39%), takes only 4.21 hours to train a model with 70*k* iterations and has a mean computation time of 0.059 seconds per each test frame. While the improvement with our multimodal approach over Faster R-CNN seems small, our architecture has repeatedly and consistently scored higher accuracy for each experiment set. Thus, we believe using a multimodal approach has a stable improvement over the architecture proposed by Ren *et al.* [8], however there might be room for improvement such as pre-training on similar flow data and transferring its weights to initialize our model. This could help our architecture refine its improvement over single modality approach. In order to test the efficiency of using Region Proposal Network to generate object region proposals, we experiment with an alternative region proposal method. Among the most popular region proposal methods; Selective Search takes about two seconds per image and generates higher quality proposals while EdgeBoxes takes only 0.2 seconds per image (PASCAL VOC); however, it compromises the quality of the proposals. With initial experimentation, we have observed that the Selective Search did not produce drastically superior results compared to EdgeBoxes on our dataset. We chose to run our experiments with EdgeBoxes with the time consumption concern. The experimental results of using EdgeBoxes with Fast R-CNN network reach an Average Precision of only 20% while it takes 0.134 for detection on top of the 2 seconds to compute the region proposals per frame on our dataset. We find 40*k* iterations to give better results instead of the 70*k*. This approach takes 1.48 hours to train with 40*k* maximum iterations. We use a subset of our training images (each experiment set above 2*k* and on average 2064 images) to train a Deformable Parts Model (DPM) (voc-release4) with memory concerns. It scores an Average Precision of 76% and up to an average of 83% with bounding box regression. The mean time of object detection by DPM in a test frame is 2.3 seconds. Each of these experiments are carried with a ten-fold approach, using the same randomized video split sets for training and testing. We show sample results in Figure 6.

VI. CONCLUSION

RAS video understanding has not yet taken advantage of the recent advances in deep neural networks. In our paper, we propose an end-to-end deep learning approach for fast tool detection and localization in RAS videos. Our architecture applies a Region Proposal Network (RPN), and a multimodal convolutional network for object detection, to jointly predict objectness and localization on a fusion of image and temporal motion cues. We also introduce our dataset ATLAS Dione which provides video data of ten subjects performing six different surgical tasks on the dVSS® with proper tool annotations. Our experimental results demonstrate that our multimodal architecture is superior to similar approaches and also the conventionally used object detection methods in medical domain with an Average Precision (AP) of 91% and a mean computation time of 0.1 seconds per test frame. With our new architecture that supports multimodality, we improve the results of the architecture proposed by Ren *et al.* [8]. Although the improvement is small, our architecture has repeatedly and consistently scored higher accuracy for each experiment set. We believe that making use of the temporal motion cues; optic flow, improves the accuracy by decreasing false positives. Using a fusion of both RGB image and flow modalities make our system more stable and our detections more confident. These findings encourage using additional modalities in detection and localization of tools in RAS videos. Using the architecture proposed by Girschick *et al.* [8], it is

possible to achieve comparable results to ours in less time for training and testing, this is mainly because our architecture has two different streams of convolutions for RGB input image and flow input image. Although the convolutions are shared for RGB stream and Region Proposal Network, the flow input is convolved in a separate stream before it is fused, almost doubling the time spent for training and testing, that is excluding the time spent to compute flow images as part of preprocessing. We believe we could further improve our accuracy following the multimodal approach; pre-training on similar flow data and transferring its weights to initialize our model could help us further refine our model. Our results show that using Region Proposal Network jointly with detection network, whether it is the new multimodal architecture we propose or the one proposed by Ren *et al.* [8], dramatically improves the accuracy and reduces the computation time for detection in each frame. We believe our study and dataset will form a benchmark for future studies.

ACKNOWLEDGMENT

The authors would like to thank our interns: Lauren Samar, who is a Biomedical Engineering student at Rochester Institute of Technology, and Basel Ahmad, who is a Biomedical Sciences student at SUNY Buffalo, for manual annotation of the data. They would like to acknowledge the ten surgeons working at RPCI who participated in this IRB approved study (I228012). They use the Caffe framework by Jia *et al.* [39] for our experiments and the optical flow estimation code by Thomas *et al.* [32] for estimating the temporal motion cues in video frames.

REFERENCES

[1] M. Meadows, "Robots lend a helping hand to surgeons," *U.S. Food Drug Admin. (FDA) Consum. Mag.*, vol. 36, no. 3, pp. 10–15, May/Jun. 2002.
[2] S. Kumar, N. Ahmidi, G. Hager, P. Singhal, J. J. Corso, and V. Krovi, "Surgical performance assessment," *ASME Dyn. Syst. Control Mag.*, vol. 3, no. 3, pp. 7–10, 2015.
[3] K. A. Guru *et al.*, "Cognitive skills assessment during robot-assisted surgery: Separating the wheat from the chaff," *Brit. J. Urol. Int.*, vol. 115, no. 1, pp. 166–174, Jan. 2014.
[4] Y. LeCun *et al.*, "Backpropagation applied to handwritten zip code recognition," *Neural Comput.*, vol. 1, no. 4, pp. 541–551, 1989.
[5] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.
[6] O. Russakovsky *et al.*, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015.
[7] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2012, pp. 1–2.
[8] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Neural Inf. Process. Syst. (NIPS)*, 2015, pp. 1–14.
[9] M. Groeger, K. Arbter, and G. Hirzinger, "Motion tracking for minimally invasive robotic surgery," in *Medical Robotics*, V. Bozovic Ed. Rijeka, Croatia: InTech Education and Publishing, 2008, pp. 117–148.
[10] G.-Q. Wei, K. Arbter, and G. Hirzinger, "Automatic tracking of laparoscopic instruments by color coding," in *Proc. 1st Joint Conf. Comput. Vis., Virtual Reality Robot. Med. Med. Robot. Comput.-Assist. Surgery (CVRMed-MRCAS)*, 1997, pp. 357–366.
[11] A. Krupa *et al.*, "Autonomous 3-D positioning of surgical instruments in robotized laparoscopic surgery using visual servoing," *IEEE Trans. Robot. Autom.*, vol. 19, no. 5, pp. 842–853, Oct. 2003.
[12] X. Zhang and S. Payandeh, "Application of visual tracking for robot-assisted laparoscopic surgery," *J. Robot. Syst.*, vol. 19, no. 7, pp. 315–328, Jul. 2002.
[13] A. Reiter, T. Zhao, and P. K. Allen, "Appearance learning for 3D tracking of robotic surgical tools," *Int. J. Robot. Res.*, vol. 33, no. 2, pp. 342–356, 2014.
[14] S. Kumar, M. S. Narayanan, P. Singhal, J. Corso, and V. Krovi, "Product of tracking experts for visual tracking of surgical tools," in *Proc. IEEE Int. Conf. Autom. Sci. Eng. (CASE)*, Aug. 2013, pp. 480–485.
[15] X. Du *et al.*, "Combined 2D and 3D tracking of surgical instruments for minimally invasive and robotic-assisted surgery," *Inf. Process. Comput.-Assist. Interventions*, vol. 11, no. 6, Jun. 2016.
[16] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.
[17] D. H. Ballard, "Generalizing the Hough transform to detect arbitrary shapes," *Pattern Recognit.*, vol. 13, no. 2, pp. 111–122, 1981.
[18] M. Allan *et al.*, "Image based surgical instrument pose estimation with multi-class labelling and optical flow," in *Proc. 18th Int. Conf. Med. Image Comput. Comput. Assist. Intervention, (MICCAI)* Munich, Germany, Oct. 2015, pp. 331–338.
[19] A. Reiter, P. K. Allen, and T. Zhao, "Marker-less articulated surgical tool detection," *Comput. Assist. Radiol. Surgery*, Pisa, Italy, Jun. 2012.
[20] R. Sznitman, C. Becker, and P. Fua, "Fast part-based classification for instrument detection in minimally invasive surgery," in *Proc. 17th Int. Conf. Med. Image Comput. Comput. Assist. Interventions (MICCAI)*, pp. 692–699, 2014.
[21] P. Felzenszwalb, D. McAllester, and D. Ramanan, "A discriminatively trained, multiscale, deformable part model," in *Proc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2008, pp. 1–8.
[22] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 9, pp. 1627–1645, Sep. 2010.
[23] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2005, pp. 886–889.
[24] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2014, pp. 818–833.
[25] R. Girshick, "Fast R-CNN," *Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1440–1448.
[26] Y. Gao *et al.*, "JHU-ISI gesture and skill assessment working set (JIGSAWS): A surgical activity dataset for human motion modeling," *5th Workshop Modeling Monitoring Comput. Assist. Interventions (M2CAI)*, Boston, MA, USA, 2014.
[27] A. P. Twinanda *et al.*, "EndoNet: A deep architecture for recognition tasks on laparoscopic videos," *IEEE Trans. Med. Imag. (TMI)*, Jan. 2017, pp. 86–97.
[28] A. P. Stegemann *et al.*, "Fundamental skills of robotic surgery: A multi-institutional randomized controlled trial for validation of a simulation-based curriculum," *Urology*, vol. 81, no. 4, pp. 767–774, 2013.
[29] P. Dollár, C. Wojek, B. Schiele and P. Perona, "Pedestrian detection: A benchmark," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2009, pp. 304–311.
[30] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn and A. Zisserman, "The pascal visual object classes (VOC) challenge," *Int. J. Comput. Vision*, vol. 88, no. 2, pp. 303–338, 2010.
[31] S. E. Dreyfus and H. L. Dreyfus, "A five-stage model of the mental activities involved in directed skill acquisition," *Oper. Res. Center*, Cambridge, MA, USA, Tech. Rep., Feb. 1980.
[32] B. Thomas, A. Bruhn, N. Papenberg, and J. Weickert, "High accuracy optical flow estimation based on a theory for warping," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2004, pp. 25–36.
[33] J. A. Baxter, "A model of inductive bias learning," *J. Artif. Intell. Res.*, vol. 12, no. 1, pp. 149–198, 2000.
[34] R. Caruana, "Multitask learning," *Mach. Learn.*, vol. 28, no. 1, pp. 41–75, 1997.
[35] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2014, pp. 580–587.
[36] C. L. Zitnick and P. Dollár, "Edge boxes: Locating object proposals from edges," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2014, pp. 391–405.
[37] J. R. R. Uijlings, K. E. A. Van De Sande, T. Gevers, and A. W. M. Smeulders, "Selective search for object recognition," *Int. J. Comput. Vis.*, vol. 104, no. 2, pp. 154–171, 2013.
[38] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2014, pp. 346–361.
[39] Y. Jia *et al.*, "Caffe: Convolutional architecture for fast feature embedding," in *Proc. ACM Int. Conf. Multimedia*, 2014, pp. 675–678.