Autonomous Robotic Surgery with Video Informatics in a Simulator Environment

Project Report to be submitted in Partial Fulfillment
of the Requirements for the Award of the Degree of

Bachelor of Technology
in
Electrical Engineering

by

Soumava Paul
(16EE10056)

Under the supervision of

Dr. Debdoot Sheet

Department of Electrical Engineering
Indian Institute of Technology Kharagpur
November 2019

Abstract

THIS thesis is based on methods for autonomous robotic surgery. We aim to achieve this objective by first creating deep learning models that can predict different kinematic variables describing the motion of robotic manipulators. To test the resulting model, we propose to simulate 3 simple robotic operations - Knot-Tying, Needle-Passing and Suturing in an OpenAI gym environment using inputs from the deep learning model.

To train these models, we use the **JHU-ISI Gesture and Skill Assessment Working Set (JIGSAWS)** dataset that comprises video files of these operations being performed together with frame-wise annotations of 38 kinematic variables that describe the motion of a pair of robotic manipulators.

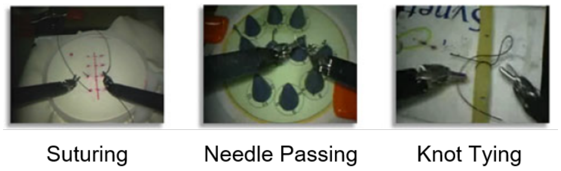
This report contains performance summaries of deep CNNs made up of a Resnet18 backbone in predicting the above sets of variables. Through our experiments, we achieve satisfactory performances on all 3 operations, with the results on Suturing being the most impressive.

Keywords: Autonomous robotic surgery, medical imaging, deep CNN, deep regression.

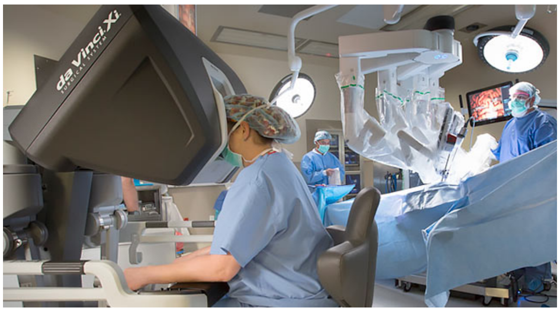
Contents

1 Introduction

ROBOTIC surgery facilitates the execution of complex surgical tasks through tiny incisions using miniaturized instruments and high-definition 3D cameras. Compared to open surgery (traditional surgery with incisions), robotic and minimally invasive surgery results in smaller incisions resulting in less pain and scarring. Patients are also less dependent on painkillers during recovery, reducing the risk of addiction. Also, blood loss being minimal, the need for blood transfusions is also eliminated. One of the most prominent examples of a robotic surgery system is the **da Vinci Surgical System** that is designed to facilitate surgery using a minimally invasive approach, and is controlled by a surgeon from a console which controls a robotic arm. However, at present these robots aren't completely sentient and cannot perform surgeries without human assistance. In this thesis, we aim at taking small steps towards this big objective using video informaics and a simulator environment. We use the JHU-ISI Gesture and Skill Assessment Working Set (**JIGSAWS**) dataset by Ahmidi et al. (2017) for testing our methods.



(a) 3 Operations in JIGSAWS dataset



(b) Da Vinci Robotic Surgery Setup

2 Prior Art

Most of the research using this dataset has been related to action recognition (Gesture Classification of 12 unique gestures across 3 operations) with input as videos or frame-wise kinematic variables. Lea et al. (2016) proposed a model for action segmentation which combines low-level spatio-temporal features with a high-level segmental classifier for joint segmentation and classification of fine-grained actions. Schulman et al. (2013) proposed a 3D warping function and trajectory optimization algorithms for automating the process of suturing.

3 Aim and Objectives

The *aim* of this thesis is to develop computational methods and techniques for automating robotic surgery on 3 operations, namely suturing, knot-tying and needle-passing in a simulator environment. The following *objectives* have been set to achieve this aim:

1. Frame-wise Regression of 38 kinematic variables of 2 sets (left and right) of robotic manipulators using Convolutional Neural Networks 4
2. Regression of Kinematic Variables using Spatio-Temporal CNNs comprising a CNN Encoder and LSTM Decoder.
3. Simulation of the 3 operations in an OpenAI gym environment

4 Work Progress and Achievements

4.1 Frame-wise Regression of kinematic variables using CNNs

4.1.1 Methodology

We use a pretrained ImageNet model - Resnet18(He et al. (2016)) for frame-wise regression of the kinematic variables. Input frames are resized to 224x224 so that the pre-trained weights can be fine-tuned. All models are trained with *MSE Loss* and L2 regularization is used for generalization.

4.1.2 Train-Val-Test Split

- There is no official train-test split provided with the dataset. We split the data based on the skill levels of the operator. Videos belonging to skill levels 1,8,4,5 are assigned to the train set, 2,7 to validation set and 3,6 to test set.

4.1.3 Depth Map creation using SGM Algorithm

We use the popular SGM algorithm (Hirschmuller (2007)) introduced in 2005 to calculate depth maps per frame using images of the left and right cameras. As shown below, the algorithm works best for image pairs from Knot-Tying and poorly for Needle-Passing. The results for Suturing are more or less satisfactory. See figures 1, 2 and 3 for example depth map calculations.

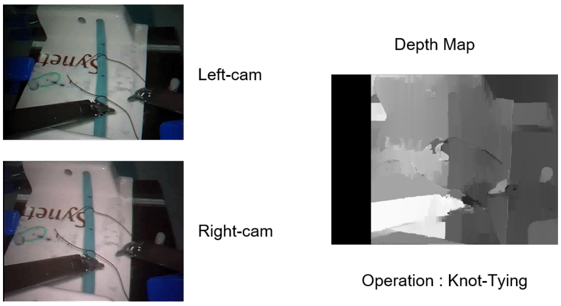


Figure 1: Left, Right Images and Depth Map for Knot-Tying

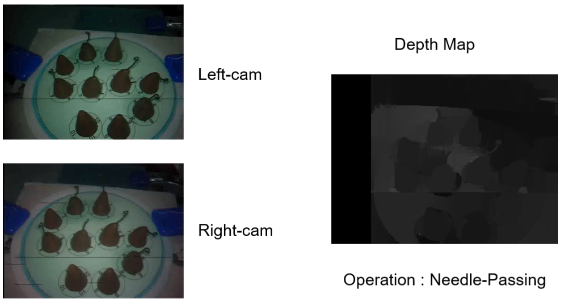


Figure 2: Left, Right Images and Depth Map for Needle-Passing

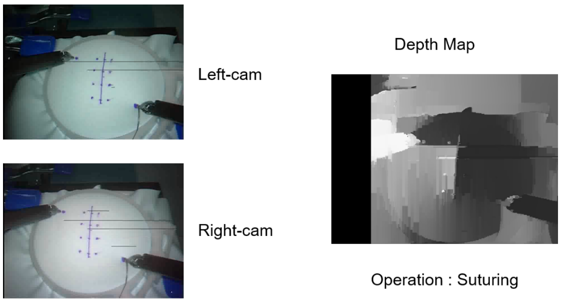


Figure 3: Left, Right Images and Depth Map for Suturing

4.1.4 Pre-processing Techniques

Since the frames are of dimensions 480x640, we resize them to 224x224 keeping this aspect ratio in mind. This introduces some black patches at the top and the bottom of the image. Also, inspired by the approach of Eitel et al. (n.d.), we transform the grayscale depth maps to the **jet** colour space. Some examples are shown in Figure 4.

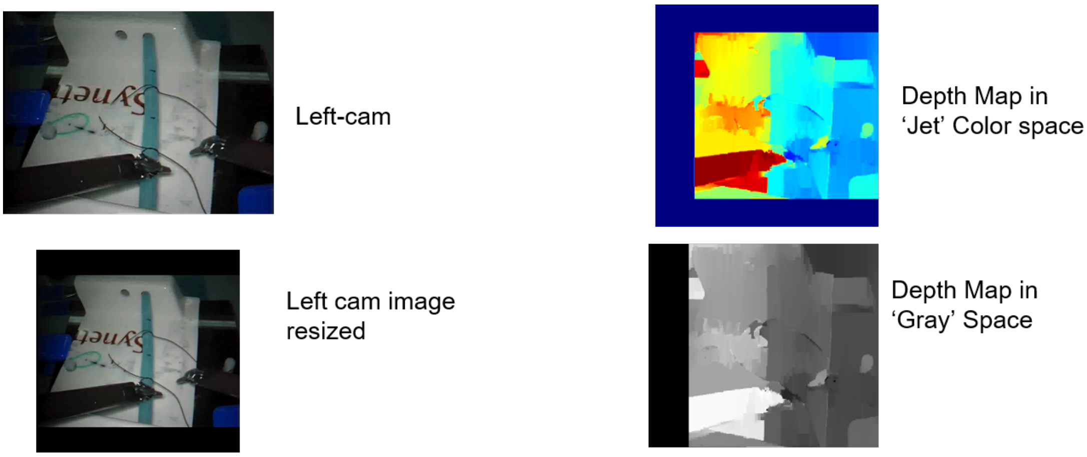


Figure 4: Pairs of some images and their resized versions

4.1.5 RGBD-CNN

Following the work of Eitel et al. (n.d.), we use an architecture composed of two separate CNN processing streams – one for each of the modalities – RGB and depth which are consecutively combined later using a fusion layer. This model is trained in an end-to-end manner after training the two sets of networks separately and saving their weights.

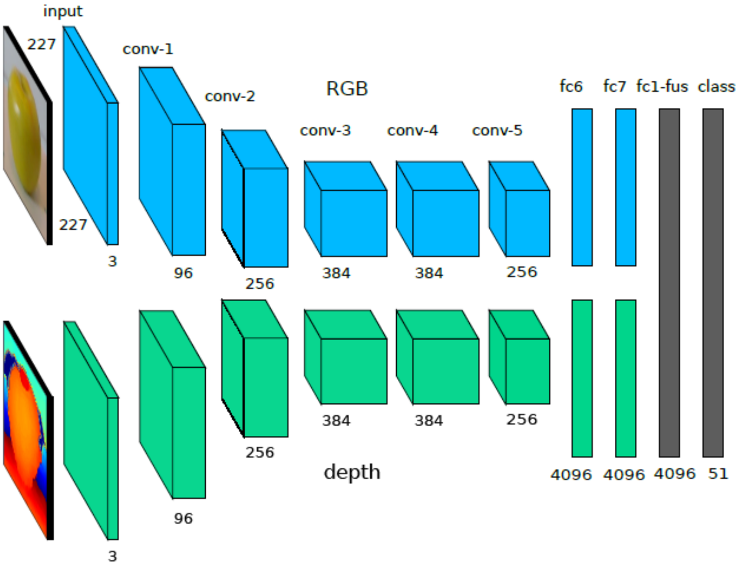


Figure 5: RGBD-CNN Architecture introduced by Eitel et al. (n.d.)

4.1.6 Dataset Description

1. **Name** - JHU-ISI Gesture and Skill Assessment Working Set (JIGSAWS)
2. **Operations** - 3: Suturing, Needle-Passing and Knot-Tying
3. **Videos** - 78 (Suturing) + 56 (Needle-Passing) + 72 (Knot-Tying) (across 8 skill levels and 2 cameras)
4. **Frame Dimensions** – 480x640
5. **Video Duration** – roughly 1 – 2.5 minutes
6. **Kinematic Variables** – Total 76 - 38 on master and another 38 on slave side. Some examples are cartesian positions (x, y, z), gripper angle, velocity (x, y, z))

4.1.7 Metrics

- R^2 score and Explained Variance Score (EVS)

$$R^2(y, \hat{y}) = 1 - \frac{\sum(y_i - \hat{y}_i)^2}{\sum(y_i - \bar{y}_i)^2}$$

$$evs(y, \hat{y}) = 1 - \frac{Var(y - \hat{y})}{Var(y - \bar{y})}$$

where y, ŷ are target and estimated outputs respectively. Both metrics have a best possible score of 1.0.

4.1.8 Results

In table 1, we report R^2 and EVS scores obtained across 3 operations using RGB and depth map units, both separately and together. The best results are obtained for Suturing.

Table 1: Performance (R^2 and EVS scores) using Baseline models and RGBD CNN.

Method	Knot-Tying	Needle-Passing	Suturing
Left Capture	R2 : -0.1574 EVS : 0.0598	R2 : **-0.1742** EVS : **0.0205**	R2 : **0.4099** EVS : **0.4259**
Right Capture	R2 : -0.2302 EVS : 0.043	R2 : -0.2486 EVS : -0.0372	R2 : 0.3215 EVS : 0.383
Depth	R2 : **0.0637** EVS : **0.1355**	R2 : -0.2742 EVS : -0.0352	R2 : 0.2781 EVS : 0.3131
Left Capture + Depth	R2 : -0.0205 EVS : 0.0493	R2 : -0.1841 EVS : -0.0049	R2 : 0.2995 EVS : 0.3229
Right Capture + Depth	R2 : 0.0156 EVS : 0.0926	R2 : -0.2896 EVS : -0.072	R2 : - EVS : -

4.2 Spatio-Temporal CNN comprising a CNN Encoder and LSTM Decoder

A CNN LSTM (Hochreiter and Schmidhuber (1997)) model can be used to provide some context while regressing the kinematic variables of a particular frame. CNN features of some prior frames (say 10-15) would provide the context and these along with the input frame would be decoded through an LSTM to regress kinematic variables of the current frame. Proposed model architecture is shown below:

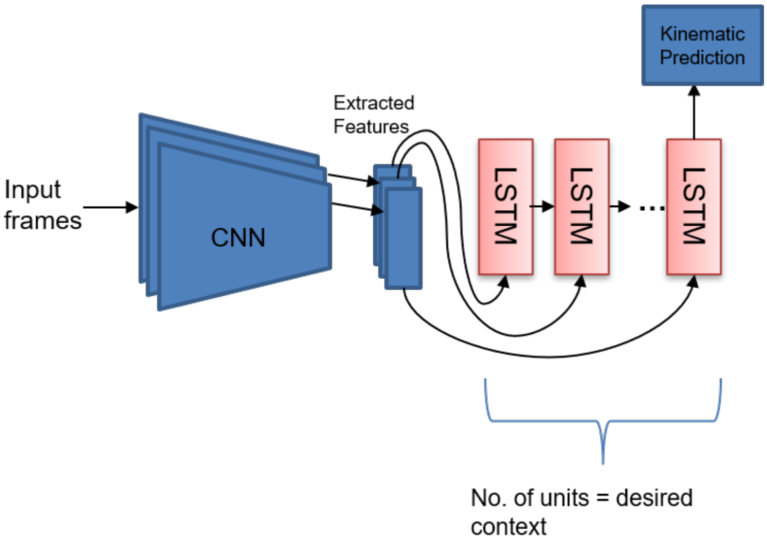


Figure 6: CNN-LSTM Architecture

4.3 Simulation of Operations in OpenAI Gym

In this part of the thesis, we aim at creating and simulating models in an **OpenAI Gym** environment that can describe the motion of the robotic manipulators using inputs from a deep learning models that will regress all associated kinematic variables. This part of the project will be performed in the following semester.

References

Ahmidi, N., Tao, L., Sefati, S., Gao, Y., Lea, C., Haro, B. B., Zappella, L., Khudanpur, S., Vidal, R. and Hager, G. D. (2017). A dataset and benchmarks for segmentation and recognition of gestures in robotic surgery, *IEEE Transactions on Biomedical Engineering* **64**(9): 2025–2041.

Eitel, A., Springenberg, J. T., Spinello, L., Riedmiller, M. and Burgard, W. (n.d.). Multimodal deep learning for robust rgb-d object recognition, *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, pp. 681–687.

He, K., Zhang, X., Ren, S. and Sun, J. (2016). Deep residual learning for image recognition, *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778.

Hirschmuller, H. (2007). Stereo processing by semiglobal matching and mutual information, *IEEE Transactions on pattern analysis and machine intelligence* **30**(2): 328–341.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory, *Neural computation* **9**(8): 1735–1780.

Lea, C., Reiter, A., Vidal, R. and Hager, G. D. (2016). Segmental spatiotemporal cnns for fine-grained action segmentation, *European Conference on Computer Vision*, Springer, pp. 36–52.

Schulman, J., Gupta, A., Venkatesan, S., Tayson-Frederick, M. and Abbeel, P. (2013). A case study of trajectory transfer through non-rigid registration for a simplified suturing scenario, *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, IEEE, pp. 4111–4117.