SuPer Deep: A Surgical Perception Framework for Robotic Tissue Manipulation using Deep Learning for Feature Extraction

Jingpei Lu, Ambareesh Jayakumari, Florian Richter, Yang Li and Michael C. Yip

Abstract— Robotic automation in surgery requires precise tracking of surgical tools and mapping of deformable tissue. Previous works on surgical perception frameworks require significant effort in developing features for surgical tool and tissue tracking. In this work, we overcome the challenge by exploiting deep learning methods for surgical perception. We integrated deep neural networks, capable of efficient feature extraction, into the tissue reconstruction and instrument pose estimation processes. By leveraging transfer learning, the deep learning based approach requires minimal training data and reduced feature engineering efforts to fully perceive a surgical scene. The framework was tested on three publicly available datasets, which use the da Vinci Surgical System, for comprehensive analysis. Experimental results show that our framework achieves state-of-the-art tracking performance in a surgical environment by utilizing deep learning for feature extraction.

I. INTRODUCTION

In the field of health care and surgery, automation is on the horizon due to advancements in robotics. Improved patient outcomes are being achieved through increased precision in tissue manipulation and the development of minimally invasive robotics [1]. One avenue of research in automation using these platforms is through the advancements of control algorithms to move towards autonomy [2], [3]. These algorithms typically aim to automate specific surgical subtasks such as suturing [4], cutting [5], and multilateral debridement [6]. Another development is assistance to the teleoperating surgeon in real-time through virtual fixtures to avoid critical areas [7], augmented reality indicators [8], and motion scaling for finer control near tissue [9].

To utilize these automation endeavors in a real surgical scene, accurate perception of the environment and the agents is essential. There are two major challenges: tracking of the surgical tool to control and localize it in the camera frame, and tracking of the deformable environment for the surgical tool to plan and interact with. While these two problems have been solved outside of surgical robotics [10], [11], the domain-specific challenges are the narrow field of view endoscopes, poor lighting conditions, and the requirement of very high accuracy [12].

The surgical tool tracking community has largely focused on developing feature detection algorithms to update the pose of the surgical tool [12]. The algorithms need to be robust to the poor lighting conditions and the highly reflective tool surfaces. Examples of recent work include using the Canny edge detector for silhouette extraction [13], online template matching [14], and classified features using classical image features such as the spatial derivatives [15] [16]. Deep neural networks have also achieved promising results in feature tracking for surgical tools [17] [18], but utilizing them for full 3D pose estimation still remains unexamined.

Simultaneously, efforts in tissue tracking have focused mainly on adaptations of 3D reconstruction techniques such as SurfelWarp for deformable tracking [11]. The lack of directly measurable depth information in endoscopes is a significant challenge in the adaptation. Hence, the common approach is to work with stereoscopic endoscopes and use stereo reconstruction techniques such as Efficient Large-Scale Stereo Matching (ELAS) to generate depth images [19]. From this depth estimation, deformable tracking techniques can be applied [20], [21]. Other tissue tracking techniques include tracking key-point features and registration [22] and dense SLAM methods, which use image features to localize the endoscope [23], [24].

A common theme across these two challenges is the need for high-quality image features. Surgical tool tracking mainly focuses on developing detectors for tool features,
and recent works in tissue tracking have highlighted depth reconstruction from stereo matching as the most significant bottleneck [20], [25]. Deep learning has the advantage of learning features, which will eliminate the need for feature engineering. However, deep learning previously has not been a front runner in surgical perception due to the lack of large quantities of high-quality medical and surgical data [26].

In this paper, we use state-of-the-art deep learning techniques that require minimal training data to explore its application in surgical perception. Our contributions can be summarized as follows:

1) Using deep learning for high quality and robust surgical tool feature extraction,
2) Applying deep learning techniques to stereo matching for accurate, precise, and dense depth estimation,
3) Complete integration of the above into our previously developed Surgical Perception (SuPer) framework [25] to fully perceive the entire surgical scene - SuPer Deep.

Experiments were run on a tissue manipulation dataset we previously released [25], as shown in Fig. 1 and two other publicly available datasets collected from the da Vinci® Surgical System, to evaluate the tool tracking and tissue reconstruction performances individually.

II. METHODOLOGY

A. The Surgical Perception Framework

Our surgical perception framework, as shown in Fig. 2, provides geometric information about the entire surgical scene, including the robotic agent and the deforming environment. Two deep neural networks are pipelined into our framework for feature extraction. The Pyramid Stereo Matching Network (PSMNet) [27] finds and matches features for stereo reconstruction, and DeepLabCut [28] detects point features for surgical tool tracking - both introduced in the following sections.

To reconstruct the surgical environment, we generated the point cloud by fusing the estimated depth maps, and a model-free tissue tracker is employed to track the deformation. The pose of the surgical tool is estimated using a model-based tracker that utilizes a kinematic prior and fusing the encoder readings with the 2D observation from the images. To efficiently combine the two separate trackers, a mask of the surgical tool is generated based on the surgical tool pose estimation and removed from the depth map given to the deformable tissue tracker. Finally, we combine the point cloud and surgical tools into the camera frame to capture the surgical scene.

B. Surgical Tool Tracking

To localize the surgical tools on image frames, we employed the deep neural network from DeepLabCut [28] for point feature detection on the surgical tool. Specifically, the deep neural network consists of variations of Deep Residual Neural Networks (ResNet) [29] for feature extraction and deconvolutional layers to up-sample the feature maps and produce spatial probability densities. The output estimation for each point feature is represented as a tuple \((h^i, \rho^i)\), where \(h^i \in \mathbb{R}^2\) is the image coordinate of the \(i\)-th feature and \(\rho^i \in \mathbb{R}\) is the corresponding confidence score. The deep neural network was fine-tuned with few training samples to adapt to surgical tool tracking. The samples were hand-labeled using the open-source DLC toolbox [30]. Fig. 3 shows examples of point features that were detected on surgical instruments.

To track the surgical tool in 3D space, the 2D detections are combined with the encoder readings from the surgical robot, and a particle filter is applied for estimation. The typical formulation used in the surgical tool tracking community is to track the error of the transform between the camera and the robotic base [14], [15] also known as hand-eye. This error
is caused by the use of large stationary set-up joints, with inaccurate encoders, to position the endoscope and surgical tool base. Calibration of hand-eye has also been shown to be of insufficient accuracy [31]. Let the error be defined as $T_b^i(\omega,b) \in SE(3)$, which is parameterized by an axis-angle vector $\omega \in \mathbb{R}^3$, and a translational vector $b \in \mathbb{R}^3$. To estimate $\omega$ and $b$ at time $t+1$ given time $t$, a zero-mean Gaussian noise is assumed to model the uncertainty. Therefore, the motion model for the particle filter is defined as:

$$\begin{align*}
[\omega_{t+1},b_{t+1}]^T & \sim \mathcal{N}([\omega_t,b_t]^T, \Sigma_{\omega,b}) \\
(1)
\end{align*}$$

where $\Sigma_{\omega,b}$ is the covariance matrix.

Using this model, a detected feature point can be projected to the image plane by being first transformed by the kinematic chain and then the corrected hand-eye. More specifically, feature point $i$ on the $j_i$-th link $p_{j_i} \in \mathbb{R}^3$ is projected onto the image plane by:

$$\begin{align*}
\bar{m}_i(\omega,b) &= \frac{1}{s} K T_{c_i}^{-1} T_{b_i}^{-1}(\omega,b) \prod_{i=1}^{j_i} T_{i-1}(\theta_i) \bar{p}_{j_i} \\
(2)
\end{align*}$$

where $T_{c_i}(\theta_i) \in SE(3)$ is the $i$-th joint transform generated by joint angle $\theta_i$, $T_{b_i}$ is the initial hand-eye transform from the set-up joints or calibration, $K$ is the intrinsic camera matrix, and $s$ is the scaling factor that constraints the point on the image plane. Note that $\gamma$ denotes the homogeneous representation of a point (e.g. $\bar{p} = [p_1,1]^T$).

From here, an observation model can be defined by relating the predicted feature location with the detected features. Given a list of $N$ observations, $(h_{t+1}, \rho_{t+1})$, the observation model is defined to be:

$$\begin{align*}
P(h_{t+1}, \rho_{t+1} | \omega_{t+1}, b_{t+1}) & \propto \sum_{i=1}^{N} \rho_i \gamma^{\frac{1}{2}||h_{i+1} - \bar{m}(\omega_{t+1}, b_{t+1})||^2} \\
(3)
\end{align*}$$

where $\gamma$ is a tuning parameter.

Since the deep neural network performs end-to-end 2D feature detection, the only parameters that need to be tuned for tool tracking are $\gamma$ and $\Sigma_{\omega,b}$. Furthermore, the feature detection from the deep neural network does not rely on the tracked states, which is a common technique in surgical tool tracking [14] and can lead to detrimental results when the tracking begins to fail. Finally, the deep neural network used here requires very few training images, as shown in the results; lack of data has been an issue in tool feature point detection [16].

C. Deformable Tissue Reconstruction

The depth map is used as an observation in tissue reconstruction, which is typically estimated using stereo reconstruction algorithms. We utilize the PSMNet [27] for stereo matching due to its ability to infer feature matching, even with minimal training data.

The deep neural network utilizes spatial pyramid pooling (SPP) and dilated convolution kernels to enlarge the receptive fields and extract region-level features of different scales. Matching is estimated by using a 3D convolutional neural network to regularize the cost volume, which is the concatenation of the extracted left and right feature maps. The output of the network is then upsampled back to the original image resolution with the estimated disparity of each pixel, which results in a dense depth image. After computing the disparity map $\hat{d}$, the depth map $D$ is obtained by the triangulation method

$$D = \frac{bf}{\hat{d}}$$

where $b$ is the distance between two cameras, and $f$ is the focal length, which can be obtained from the camera calibration. To estimate the tissue deformation, the depth map $D$ is then passed into a deformable tissue tracker without smoothing or filtering. The deformable tissue tracker previously developed by [25] is employed to fuse the depth maps, recover the deformation of the environment, and it outputs the optimized model as a point cloud. Readers may refer to [25] for more details on the deformable tissue tracker.

Since the depth map is fused into the tissue model without post-processing, no parameters need to be tuned beside the deformable tissue tracker, which is a significant improvement compared to previous tissue reconstruction techniques. Additionally, due to lack of ground truth data for depth images in surgical scenes, no additional training is applied to the PSMNet, which is pre-trained on publicly available benchmark datasets. The results from our experiments show that even without surgical scene-specific training data the network can generalize to achieve excellent performance.

III. EXPERIMENTS

We evaluated the proposed framework on three open-source datasets for multiple tasks addressing the performance of the surgical tool pose estimation and deformable tissue reconstruction. We compared it with the state-of-the-art methods for analysis. The experiments were conducted on two identical computers, each containing an Intel® Core™ i9-7940X Processor and NVIDIA’s GeForce RTX 2080.

A. Datasets and Evaluation Metrics

The Surgical Perception (SuPer) dataset mentioned in [25] is a recording of a repeated tissue manipulation experiment using the da Vinci® surgical robot. The dataset

*https://sites.google.com/ucsd.edu/super-framework/home
consists of a raw stereo endoscopic video stream and encoder readings from the surgical robot with ground-truth labels for the tool tracking and tissue tracking tasks. The testing samples include hand-labeled surgical tool masks for tool segmentation and 20 feature points with labeled positions on the deformable tissue through time, for tissue tracking. The tool tracking performance was evaluated by calculating the Intersection-Over-Union (IoU, Jaccard Index) for the rendered tool masks, which are based on estimated tool poses. The tissue tracking performance was evaluated by calculating the re-projection error of the key points, where we initialized the key points on the reconstructed tissue model in the first frame and re-projected back to the image plane for the ongoing frames.

The Hamlyn Centre Video Dataset [32] was used to evaluate the performance of deformable tissue reconstruction. It includes two video sequences of silicone heart phantom deforming with cardiac motion and consists of ex-vivo endoscopic stereo videos (resolution: 360 × 288) with depth information generated from CT scans. The re-projected depth map is the projection of the entire reconstructed point cloud to the image plane, with each pixel containing a depth value. We calculated the per-pixel root-mean-square (RMS) error of the re-projected depth map for every image:

\[
\sqrt{\frac{1}{N_p} \sum_{i,j} (\hat{d}_{i,j} - d_{i,j})^2}
\]  

(5)

where \(i, j\) is the pixel position, \(\hat{d}\) is the re-projected depth value, \(d\) is the actual depth value, and \(N_p\) is the total number of pixels for each image. We also reported the percentage of the valid (non-zero) pixels of the re-projected depth map.

The da Vinci tool tracking dataset used in [14] consists of a stereo video stream and the corresponding kinematic information of the da Vinci® surgical robot. The dataset is used to evaluate surgical tool feature detection and pose estimation. Note that the SuPer dataset has painted markers, and hence this additional experiment ensures that DeepLabCut learns surgical tool point features and is not dependent on colored markers. The performance of feature detection is evaluated by calculating the \(L^2\) norm of the error in pixels, for the \(i\)-th feature:

\[
\frac{1}{N} \sum_{n=1}^{N} ||h_i^n - t_i^n||_2
\]  

(6)

where \(N\) is the total number of test images, \(h_i^n\) is the predicted feature point location, and \(t_i^n\) is the ground truth feature point location in the \(n\)-th image. We experiment with varying amounts of hand-labeled training data to illustrate the efficiency of transfer learning. Due to lack of ground-truth data for pose estimation, we only provide the qualitative results for this part of the analysis.

B. Implementation Details

1) Surgical Tool Pose Estimation: For the SuPer dataset, the images were resized to (960, 540) before passing to DeepLabCut for feature detection. The weights of DeepLabCut were pre-trained on ImageNet and fine-tuned by training on only 50 hand-labeled images for 7100 iterations. For the particle filter, we used 1000 particles with bootstrap approximation on the prediction step and stratified resampling when the number of effective particles dropped below 500. For initialization, the parameterized hand-eye error is set to \([\omega_0^T, b_0^T]^T = [0, 0, 0, 0, 0, 0]^T\) with \(\Sigma_0 = \text{diag}(0.005, 0.005, 0.005, 0.025, 0.025, 0.025)\). For the motion model, the covariance is \(\Sigma_{\omega,b} = 0.1 \times \Sigma_0\), and the \(\gamma\) is set to 0.1 for the observation model.

2) Deformable Tissue Reconstruction: For depth map estimation, the raw stereo images were rectified, undistorted,
Fig. 5: Qualitative results of surgical tool tracking. The top row shows the DeepLabCut prediction overlaid on the real images. The bottom row shows an Augmented Reality rendering of the surgical tool [33] on top of the real images. The renderings are best viewed in color due to near-perfect overlap.

Fig. 6: A comparison of resulting depth maps from different stereo matching algorithms.

| Method    | Video 1  | Video 2  |
|-----------|----------|----------|
|           | RMSE     | Perc. valid | RMSE     | Perc. valid |
| stereoBM  | 23.26    | 0.565     | 34.02    | 0.523      |
| stereoSGBM| 16.84    | 0.713     | 24.71    | 0.683      |
| SuPer     | 16.12    | 0.716     | 22.05    | 0.719      |
| SuPer Deep| 5.64     | 0.940     | 8.32     | 0.939      |

TABLE I: A comparison of the re-projection depth map of the Hamlyn Validation Dataset. The percentage of valid pixels and per-pixel RMS error are measured.

and resized to (640, 480) before being passed into PSMNet for stereo matching. Due to the lack of task-specific datasets for surgical environments, the weights of PSMNet were trained on the KITTI 2015 dataset, with stacked hourglass modules enabled. The resulting depth map was fused into the tissue model after subtracting the rendered tool mask, which is dilated by 5 pixels.

IV. RESULTS

Qualitative results of the environment mapping on the SuPer dataset are presented in Fig. 4. As highlighted in the figures, SuPer Deep provides a larger field of view of the unstructured environment while preserving better details on the reconstruction. In comparison, in the results of the original SuPer framework, detailed information is lost due to filtering and smoothing in the stereo reconstruction process.

1) Deformable Tissue Tracking: Using the Hamlyn Centre Video Dataset, the deformable tissue reconstruction results were compared by combining popular stereo reconstruction algorithms with the deformable tissue tracker. The stereoBM† and stereoSGBM [34] algorithms are implemented using OpenCV package. SuPer utilizes the ELAS for stereo matching, which is implemented using the open-source library [19]. We calculated the average per-pixel RMS error on the depth maps, and the quantitative results are shown in Table II. Our method achieves the lowest per-pixel RMS error with the highest percentage of the valid pixel, which confirms the observations from the environment mapping results in Fig. 4. To compare the performances of the stereo matching algorithms, we visualize the estimated depth maps from each in Fig. 6. It is evident that PSMNet provides the best depth observation for the deformable tissue reconstruction, by providing more dense and consistent matches.

For the tissue tracking task on the SuPer dataset, the quantitative results are shown in Table II, where our method

| Method    | Error          |
|-----------|----------------|
| SURF      | 0.1746 ± 0.1110|
| SuPer     | 0.0337 ± 0.0139|
| SuPer Deep| 0.0299 ± 0.0130|

TABLE II: The re-projection error comparison of 20 labeled points from the SuPer dataset. The error is the mean and standard deviation of all 20 points and is presented as the percentage of the image resolution (640 by 480).

†https://docs.opencv.org/4.2.0/d9/dba/classcv_bilateral_filter.html
was compared with the original SuPer [25] and SURF [35] implementations. The SURF implementation matches the key points in the ongoing frames with the descriptors of the points, initialized in the first frame. While deformation tracking methods generally perform better than the feature tracking methods, our method performs the best by utilizing the deep neural network for stereo matching.

2) Surgical Tool Tracking: Fig. 7 shows the feature detection performance of the DeepLabCut with varying numbers of training samples. By leveraging transfer learning, DeepLabCut is able to achieve high performance on detecting surgical tool features using a few training samples.

For the tool tracking task, SuPer Deep achieved 91.0% mean IoU on the SuPer tool segmentation task, which is a significant improvement on the original method (SuPer: 82.8%). Notably, SuPer Deep does marker-less tool tracking while the former utilizes painted markers. The qualitative results of the tool tracking are presented in Fig. 8 where we experimented with our tool tracker on both the SuPer dataset and the da Vinci tool tracking dataset. In the visualization, the Augmented Reality rendering from the estimated tool pose produces a near-perfect overlap with the tool on real images.

V. DISCUSSION

The experimental results show SuPer Deep successfully capturing surgical environments through deep learning, with excellent performances in both surgical tool tracking and tissue tracking tasks. By utilizing deep neural networks, SuPer Deep produces more consistent disparity maps and achieves accurate tool pose estimation. The latter also helps to reduce the dilation of the tool mask, which reduces the amount of information lost. As the visualizations show in Fig. 8, SuPer Deep’s reconstruction shows the tool touching the point cloud (as opposed to just being above the point cloud).

There are occasional failures in feature detection, owing mainly to the symmetry of the tool parts, for example, the Roll_1, Pitch_1 and Yaw_2 features. As shown in Fig. 7, detecting those features are more challenging compared to other ones. The misdetections are, however, of low confidence. Hence they are handled by the probability weighting of the detected points in the observation model of the particle filter. In Fig. 8, for instance, one of the grippers of the tools is misdetected, but has substantially lower confidence; the correctly detected points have confidence scores higher than 70%. Similarly, in the second case, two features are detected on the wrong side of the shaft (they are symmetric and partially visible in this frame), but again with low confidence and hence does not throw off the pose estimation. Overall, the feature detection is robust and results in accurate perception.

VI. CONCLUSION

Deep learning has not been utilized as a major tool in surgical robotic perception, with a lack of training data pointed out as the primary bottleneck. The SuPer Deep framework, incorporating two deep neural networks as major components, shows that the challenge of insufficient data is surmountable. Using transfer learning, even on limited training data, the framework accomplishes excellent feature detection for surgical scene perception.

Currently, we believe that the major limitation of the SuPer Deep framework is its high computation power. Running multiple deep neural networks in real-time requires multiple processing units, which limits the update rates. Our experiments required two computers. Lightweight deep neural networks, like MobileNets [36], will be ideal for real-time surgical applications, if adapted without compromising on accuracy. Lack of diverse task-specific datasets is a challenge in benchmarking approaches and limits a thorough analysis of different methods. Although we released our own open-sourced SuPer dataset previously, it is still limited to a single tissue type, surgical tool, lighting condition, and task. A larger variety of datasets would enable quicker development and validation of algorithms.

For future work, we will conduct further testing of the SuPer Deep framework by collecting different surgical scene data. Another direction to pursue is surgical task automation. By using the perceived environment as feedback, controllers applied to the surgical tool will be able to accomplish tasks in unstructured, deforming surgical environments. A self-supervised learning approach would also drive advancements in surgical automation. As human-labeled features are imperfect, utilizing synthetic data and domain randomization [37] will have benefits like reducing labeling efforts, further optimize neural networks, and learn better features to improve the robustness of perception algorithms.
REFERENCES

[1] G. H. Ballantyne and F. Moll, “The da vinci telerobotic surgical system: the virtual operative field and telepresence surgery,” *Surgical Clinics*, vol. 83, no. 6, pp. 1293–1304, 2003.

[2] M. Yip and N. Das, “Robot autonomy for surgery,” in *Encyclopedia of Medical Robotics*, ch. 10, pp. 281–313, World Scientific, 2017.

[3] F. Richter, R. K. Orosco, and M. C. Yip, “Open-sourced reinforcement learning environments for surgical robotics,” *arXiv preprint arXiv:1903.02900*, 2019.

[4] R. C. Jackson and M. C. Çavusoğlu, “Needle path planning for autonomous robotic surgical suturing,” in *ICRA*, pp. 1669–1675, 2013.

[5] T. Weyand, M. Andreetto, and H. Adam, “Mobilenets: Efficient convolutional neural networks for mobile vision applications,” *arXiv preprint arXiv:1704.04861*, 2017.

[6] Y. Li, J. Zhu, S. C. Hoi, W. Song, Z. Wang, and H. Liu, “Robust estimation of similarity transformation for visual object tracking,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, pp. 8666–8673, 2019.

[7] W. Gao and R. Tedrake, “Surfelwarp: Efficient non-volumetric single view dynamic reconstruction,” in *Robotics: Science and System*, 2018.

[8] D. Bouget, M. Allan, D. Stoyanov, and P. Jannin, “Vision-based and marker-less surgical tool detection and tracking: a review of the literature,” *Medical Image Analysis*, vol. 35, pp. 633–654, 2017.

[9] R. Hao, O. Özgünere, and M. C. Çavuşoğlu, “Vision-based surgical tool pose estimation for the da Vinci® robotic surgical system,” in *Intl. Conf. on Intelligent Robots and Systems*, IEEE, 2018.

[10] M. Ye, L. Zhang, S. Giannarou, and G.-Z. Yang, “Real-time 3D tracking of articulated tools for robotic surgery,” in *Intl. Conf. on Medical Image Computing and Computer-Assisted Intervention*, pp. 386–394, Springer, 2016.

[11] A. Reiter, P. K. Allen, and T. Zhao, “Feature classification for tracking articulated surgical tools,” in *Intl. Conf. on Medical Image Computing and Computer-Assisted Intervention*, pp. 592–600, Springer, 2012.

[12] A. Reiter, P. K. Allen, and T. Zhao, “Appearance learning for 3D tracking of robotic surgical tools,” *The International Journal of Robotics Research*, vol. 33, no. 2, pp. 342–356, 2014.

[13] T. Kurmann, P. Marquez Neila, X. Du, P. Fua, D. Stoyanov, S. Wolf, and R. Sznitman, “Simultaneous recognition and pose estimation of instruments in minimally invasive surgery,” *Medical Image Computing and Computer-Assisted Intervention - MICCAI*, vol. 2017, p. 505–513, 2017.

[14] E. Collonei, S. Moccia, X. Du, E. De Momi, and D. Stoyanov, “Deep learning based robotic tool detection and articulation estimation with spatio-temporal layers,” *IEEE Robotics and Automation Letters*, vol. 4, pp. 2714–2721, July 2019.

[15] A. Geiger, M. Roser, and R. Urtasun, “Efficient large-scale stereo matching,” in *Asian Conf. on Computer Vision*, pp. 25–38, 2010.

[16] J. Song, J. Wang, L. Zhao, S. Huang, and G. Dissanayake, “Dynamic reconstruction of deformable soft-tissue with deformable scope in minimally invasive surgery,” *RA-Letters*, vol. 3, no. 1, pp. 155–162, 2017.