

Using Monocular Images to Estimate Interaction Forces During Minimally Invasive Surgery

Ehsan Noohi, Sina Parastegari, Miloš Žefran

Abstract—The lack of haptic feedback during minimally invasive surgery can cause significant tissue damage and increase morbidity. Estimating the applied force from endoscopic images is a promising approach, especially using binocular images. However, many existing operation rooms are only equipped with monocular endoscopes, making force estimation more problematic. In this paper a new method for estimating the applied force from monocular endoscope images is proposed. The main contribution is the concept of virtual template that enables modeling of surface deformation without the knowledge of the undeformed shape. Results of the in vitro experiment with the lamb liver support the practicality and effectiveness of the proposed method.

I. INTRODUCTION

Minimally Invasive Surgery (MIS) reduces the patient recovery time and minimizes after-operation complications. But the global trend in adopting MIS shows that the risk of morbidity increases in MIS due to the lack of haptic feedback. The elevated morbidity is mainly related to the tissue damage resulting from the tool-tissue interaction in absence of haptic feedback [1]. As an attempt to equip the surgical tools with haptic feedback, Robot-assisted MIS (RMIS) is exploited. In addition to the considerably higher cost of RMIS systems, sensing the forces that are applied to the tissue remains challenging for RMIS. As of now, no practically reliable and clinically approved tactile system exists [2]. The alternative approach is to estimate the applied force.

Estimating the interaction forces during MIS exploits the property that living organs behave as elastic materials. That is, they gradually restore their original shape after the deforming force is removed. In fact, if the applied force permanently deforms an organ, damage has occurred; the surgeon should have been alarmed about the force magnitude long before such deformation occurs. The relationship between the applied force and the deformation of the object is modeled by elastic modulus, which in case of linear elastic materials reduces to Hooke's law. Conducting several experiments on seven different organs, [3] proposes a quantitative model that relates the applied force to the deformation depth.

The readily available source of information for extracting organ's deformation structure during MIS is endoscopic video. Recently equipped operation rooms take advantage of

the stereoscopic vision. There exist standard algorithms in computer vision that reconstruct the 3D surface of an object from its stereo-images. The performance of these methods relies on the precision of the point correspondence between the stereo-images. Therefore image noise (due to the smoke) and occlusions, which are inherent to MIS images, degrade the performance of 3D reconstruction of the organ's surface drastically. Interesting strategies have been proposed for enhancing the precision of the surface reconstruction, see e.g. [4]. While stereo vision introduces an acceptable solution for surface reconstruction, for many operation rooms that are using monocular images, the cost of upgrading to stereoscopic vision is problematic.

In this paper we propose a method that estimates the applied force based on the reconstructed deformation structure. There exist many works on deformation estimation from a monocular image as discussed below. These methods rely on knowing the undeformed shape of the object. In contrast, we limit ourselves to MIS-specific setting and propose the concept of a virtual template that allows us to estimate the deformation without any a-priori shape information. To experimentally evaluate our method, a PUMA robot was programmed to poke a lamb liver, while the interaction forces were recorded by a force sensor. The force signals are compared with the estimated force (generated by our algorithm) to evaluate the performance of the proposed algorithm.

II. RELATED WORK

There exist many approaches to model the deformation of an object. One classic approach, exemplified for e.g. by [5], is to consider the physical laws that govern the object deformation and to minimize the energy constraints. This approach requires prior knowledge of the object's physical parameters, which are not available for living organs inside the human body. Another trend in modeling the surface deformation is to exploit learning algorithms to capture the deformation structure for different conditions (see e.g. [6]). These methods require a rich set of training data, which is difficult to obtain in MIS applications.

Interpolation-based methods are a totally different approach that takes advantage of the regularity of the organs' surface. Given the appropriate set of control points on the object's surface, one can reconstruct the surface, using interpolation techniques (see e.g. [7]). High accuracy is achievable by proper selection of both control points and curve-fitting functions (e.g. polynomials, splines, radial basis functions). To employ this method in MIS, a sufficient number of traceable markers should be attached to the

Ehsan Noohi, Sina Parastegari and Miloš Žefran are with the Robotics Laboratory, Department of Electrical and Computer Engineering, University of Illinois at Chicago, Chicago, Illinois, 60607, USA. Email: {enoohi2,mparas2,mzefran}@uic.edu

surface of an organ during the surgery so the approach is not applicable.

Another interesting approach is template-based deformation modeling (see e.g. [8]). Assume that the initial 3D surface shape (template) is known and its 2D image (reference image) is available. If a 2D image of the deformed object is available and the matching between the feature points in this image and the reference image is also available, then the deformed surface can be reconstructed from the template. Unfortunately, the governing system of equations is very ill-conditioned and generates many similar deformed surfaces (solutions). Several different approaches have been introduced to disambiguate the solution (see e.g. [9] and [10]). Nevertheless, this approach is impractical for MIS because the template of the organ is not available.

The main difficulty of applying the above approaches to MIS is that they require prior knowledge that is not readily available. To tackle this issue, a new trend has emerged in recent years. Recovering a rigid body shape from video images by exploiting Structure-From-Motion (SFM) algorithms is a well explored approach [11]. As a natural extension to the SFM, Non-Rigid Structure-From-Motion (NRSFM) is proposed in [12]. The basic assumption in NRSFM is that the deformed surface is a linear combination of the basis shapes that needs to be determined by the algorithm. The idea is to find the best basis shapes that can minimize the projection error, while keeping temporal consistency between the frames. Theoretically, NRSFM works even without the knowledge of camera internal parameters or the camera motion model. However, each unknown aspect adds to the ambiguity of the solution, making NRSFM sensitive to the image noise and initialization [13]. In particular, image noise due to the smoke during the MIS and tool occlusions are very hard to deal with in this approach. Also, in a complex deformation, the linear subspaces introduced in NRSFM are not sufficient anymore and higher-order (quadratic) models are required [14].

III. VIRTUAL TEMPLATE

To avoid limitations of NRSFM, we propose a virtual-template-based method that takes advantage of robustness and simplicity of template-based methods. However, to avoid relying on prior knowledge (actual organ template), we introduce the concept of virtual template. The key idea is to assume that the organ surface is a smooth function and that the deformation is local. For instance, large portions of the surface of a liver can be approximated by a plane.

Figure 1 schematically depicts the application of a virtual template to a spherical object (Figure 1.a). The deformation is such that part of the sphere is pushed back and transformed to a plane (Figure 1.c). We took a plane as the virtual template for the sphere (Figure 1.b). It is natural to infer that when the sphere is transformed to a plane, a plane will transform to a spherical shape. Simulation results also show that the deformed part of the sphere pops out of the virtual template, like a half sphere (Figure 1.d). Therefore, even without knowing the actual template for the object, we can determine the deformation patch and its transformed structure. The reconstructed deformed surface will be a good approximation of the actual deformation, if the virtual

template is a local approximation of the actual template (e.g. a plane). Therefore, measuring the tool penetration inside the organ is equivalent to calculating the depth of the deformed patch. Virtual template will provide this depth with some scale ambiguity. We will discuss how to disambiguate this scale in section V.C.

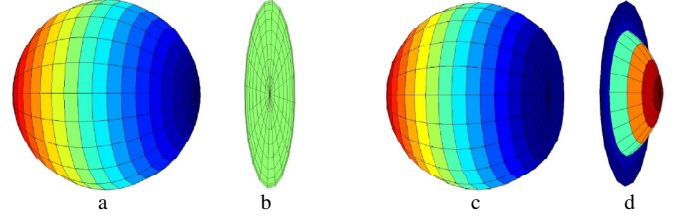


Figure 1: Simulation results of the virtual template concept. The deformation patch is determined from the virtual template: a) original object, b) virtual template of the original object, c) deformed object, d) deformed virtual template.

IV. BACKGROUND

A. Camera Inverse Model

Considering the pinhole camera model and the homogenous transformation notation, the mapping between the points in the world coordinates and the corresponding points in the camera coordinates is as follows:

$$w[u \ v \ 1]^T = K[R \ T][x \ y \ z \ 1]^T \quad (1)$$

where K represents the internal camera parameters, R and T are the extrinsic camera parameters, (x, y, z) is the position of a point in 3D world coordinates, (u, v) stands for its corresponding pixel in the 2D image coordinates and w is the z component of the projected point in the camera frame. Since the camera's pose is available in MIS (e.g. through magnetic trackers), without loss of generality we can choose the world reference frame to be the same as camera's reference frame. Therefore, $R = I_{3 \times 3}$ and $T = 0_{3 \times 1}$ and the projection mapping can be rewritten as:

$$w[u \ v \ 1]^T = K[x \ y \ z]^T \quad (2)$$

where $K = \begin{pmatrix} \gamma_x & \gamma_{xy} & -d_x \\ 0 & \gamma_y & -d_y \\ 0 & 0 & 1 \end{pmatrix}$, γ_x and γ_y are scaling factors in x and y directions, γ_{xy} is the skew between x and y axes, and d_x and d_y are x and y components of the origin of the 2D image frame in the reference frame. We can rewrite (2) as:

$$\begin{pmatrix} x/z \\ y/z \\ 1 \end{pmatrix} = K^{-1} \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \begin{pmatrix} h_x(u, v) \\ h_y(u, v) \\ 1 \end{pmatrix}. \quad (3)$$

If the object's surface is described by $S(x, y, z) = 0$, then (3) can be solved on this surface and the camera inverse model will be obtained for this object. That is, each pixel (u, v) in the image can be uniquely projected back to the corresponding point on the object's surface (x, y, z) as:

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} h_x(u, v) \\ h_y(u, v) \\ h_z(u, v) \end{pmatrix}, \quad (4)$$

Here, $z = h_z(u, v)$ is the solution of the surface function, $S(z h_x(u, v), z h_y(u, v), z) = 0$, assuming that it is solvable in terms of z . This will be generically true since the image is a projection of the object's surface.

B. Acceptable Parametric Surface Model

Let the implicit surface $S(x, y, z) = 0$ describes the object's surface. Also, assume that a 3D free-form parametric model $X(s, t) = \Gamma(s, t, C)$ describes different shapes of the same surface, where $X = (x, y, z)$ is any point on the surface, s and t are the surface parameterization variables, and $C = [c_1, c_2, \dots, c_n]$ is the set of n shape variables. The parametric model is assumed to be able to capture the deformation structure. For a model to be acceptable in our method, it should be invertible in the following three ways:

1. For a set of m independent points, X_i^* , and their corresponding parameterization variables, s_i^* and t_i^* , the set of $3m$ equations $X_i^* - \Gamma(s_i^*, t_i^*, C) = 0$ is solvable ($3m \geq n$) for the shape variables. These variables are obtained as $C = \Gamma_C(\{X_i^*, s_i^*, t_i^*\})$, where $\Gamma_C(\cdot)$ is the solution function.
2. For a given point, X^* , and a known set of shape variables, C^* , three equation $X^* - \Gamma(s, t, C^*) = 0$ are solvable for the parameterization variables. These variables are then described as $(s, t) = \Gamma_{st}(X^*, C^*)$, where $\Gamma_{st}(\cdot)$ is the solution function.
3. Let's define $Q = (x/z, y/z, 1)$ as the transformed version of the point $X = (x, y, z)$. For a set of k points X_i , their corresponding transformed points, Q_i^* (which are assumed to be independent), and their corresponding parameterization variables, s_i^* and t_i^* , are assumed to be known. The set of $3k$ equations $Q_i^* - z \Gamma(s_i^*, t_i^*, C) = 0$ should be solvable ($2k \geq n$) for the shape variables. These variables are described as $C = \Gamma_Q(\{Q_i^*, s_i^*, t_i^*\})$, in which $\Gamma_Q(\cdot)$ is the solution.

V. METHOD

A. Notations and Assumptions

To construct the deformation structure, the algorithm takes two images to work on. First one is the image of the original 3D object, before the deformation. The second one is the image of the deformed object. It is obvious that the method is well-suited for the video stream in MIS operations. Let's refer to the pixels in the reference image by (u^r, v^r) and to the pixels in the target image (obtained from the deformed object) by (u^d, v^d) .

Let's assume that the virtual template is available as $S(x, y, z) = 0$ and an acceptable surface model exists that describes the virtual template as $X(s, t) = \Gamma(s, t, C)$. Superscript r stands for the virtual template that is associated with the reference image and superscript d for the target image. Therefore, $X^r = (x^r, y^r, z^r) = \Gamma(s^r, t^r, C^r)$ and $X^d = (x^d, y^d, z^d) = \Gamma(s^d, t^d, C^d)$ describe the reference template and the target template, respectively.

Since the target template is a deformed version of the reference template, there exists a correspondence relationship between the points on both templates. For a certain point on the reference template, X^r , let's assume X^d to be the exact same point on the deformed template. An intuitive assumption for a surface model is that these two points have

the same representation in the surface parameterization space. In other words, $s^r = s^d$ and $t^r = t^d$. We will refer to this assumption as the "deformation assumption" hereafter and will assume that the acceptable surface model would not violate this assumption.

B. Deformation Reconstruction Algorithm

The deformation reconstruction algorithm requires four inputs: the reference image, I^r , the target image, I^d , the virtual template, $S^r(X) = 0$, and the parametric surface model, $\Gamma(\cdot)$. The algorithm calculates the shape variables of the deformed surface, C^d , by executing following five stages:

1. Fit the desired model Γ to the reference template S^r . In other word, calculate $C^r = \Gamma_C(\{X_k^r, s_k^r, t_k^r\})$ for a set of points on the surface, i.e. $S^r(X_k^r) = 0$. The selection of the points, X_k^r , their corresponding variables s_k^r and t_k^r , and the minimum number of required points are design options. However, constraint 1 of the model acceptability should be satisfied, see section IV.B.
2. Find a set of feature points, f_i , on the reference image, $f_i^r = (u_i^r, v_i^r)$, and their corresponding points on the target image, $f_i^d = (u_i^d, v_i^d)$. The minimum number of required feature points is related to the number of shape variables, n , see section IV.B.
3. Using camera inverse model, project the reference feature points, (u_i^r, v_i^r) , back to the 3D points on the reference virtual template, F_i^r . In other words, plug the virtual reference template $S^r(X) = 0$ and (u_i^r, v_i^r) in (4) to compute F_i^r . Note that $S^r(F_i^r) = 0$.
4. Find the parameterization variables for the feature points on the virtual template. In other words, first compute $(s_i^r, t_i^r) = \Gamma_{st}(F_i^r, C^r)$ for all feature points and then, according to the deformation assumption, obtain $s_i^d = s_i^r$ and $t_i^d = t_i^r$.
5. Extract the shape variables for the deformed template, C^d , using the feature points in the target image, (u_i^d, v_i^d) , and the associated parameterization variables, (s_i^d, t_i^d) . Considering (3), we can rewrite $Q_i^d = (x_i^d/z_i^d, y_i^d/z_i^d, 1)$ as follows:

$$Q_i^d = \begin{pmatrix} x_i^d/z_i^d \\ y_i^d/z_i^d \\ 1 \end{pmatrix} = \begin{pmatrix} h_x(u_i^d, v_i^d) \\ h_y(u_i^d, v_i^d) \\ 1 \end{pmatrix}. \quad (5)$$

Therefore, extracting the shape variables means to compute $C^d = \Gamma_Q(\{Q_i^d, s_i^d, t_i^d\})$, or equivalently to compute $C^d = \Gamma_{uv}(\{u_i^d, v_i^d, s_i^d, t_i^d\})$.

The deformed surface function $S^d(X) = 0$ is then available from the extracted model as: $X(s, t) = \Gamma(s, t, C^d)$. Figure 2 depicts a schematic, illustrating the interconnections of the steps and inputs/outputs of each step in the algorithm; along with the graphical representations of the notation. In the next section, we will discuss common practical issues that may appear in our method and will provide proper solutions.

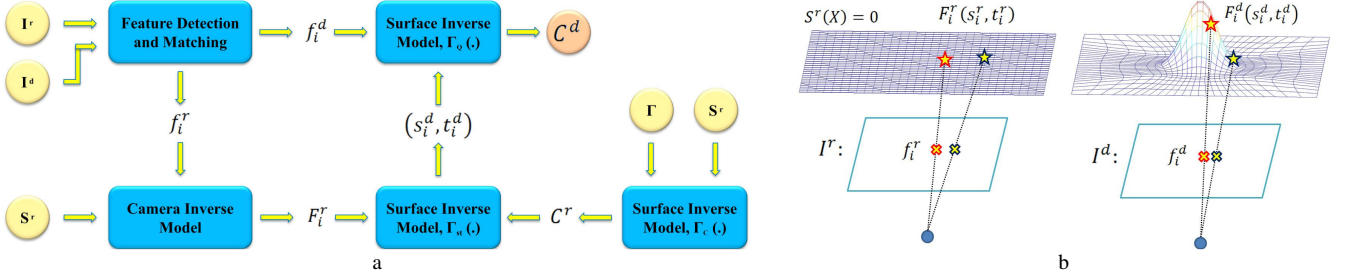


Figure 2: Illustration of the algorithm. a) The schematic of the deformation reconstruction algorithm. The circles with yellowish color are the inputs and the reddish colored circle is the output of the algorithm. Blue blocks represent different stages of the algorithm. b) The graphical representation of the notations used in the algorithm. A blue dot is the camera focal point, a cross marker indicates a pixel on an image, i.e. $f_i^r = (u_i^r, v_i^r)$ on I^r or $f_i^d = (u_i^d, v_i^d)$ on I^d , and a star is its corresponding point on the surface, e.g. point $F_i^r(s_i^r, t_i^r) = \Gamma(s_i^r, t_i^r, C^r)$ on surface $S^r(X) = 0$.

C. Discussion

As it was stated in section III, the deformed template that our algorithm constructs is a transformed version of the actual object's deformed surface. However, since the actual template is unknown, the transformation function is also unavailable. Therefore, the tool penetration depth that our algorithm calculates has an unknown relation to the actual penetration depth. To overcome this issue, we take advantage of the smoothness of the organs' surfaces in human body. If the virtual template is also a smooth surface, let's say a plane, our proposed algorithm actually finds a local approximation for the organ's surface in the vicinity of the deformation area. The approximate location of the center of the deformation patch is obtained by tracking the tool-tip. The experimental results show that this approximation works very well for a lamb liver (see section VI).

There exist some practical issues as well. As it is apparent from (3), for any pixel (u, v) in the image there are infinite points mapping to the same pixel; well-known as "scale ambiguity" in the monocular vision. This issue may affect the proposed algorithm in its fifth stage. That is, there may exist infinite number of different solutions for $C^d = \Gamma_{uv}(\{u_i^d, v_i^d, s_i^d, t_i^d\})$ and therefore infinite number of deformed surfaces. For instance, in a triangulated mesh model, if C^d is a solution (vertices), $k \cdot C^d$ is also a solution for any scalar value of $k > 0$.

An easy approach to resolve this issue is to constraint the deformed surface to pass through a certain point. However, since none of the points on the surface of the deformed object is available, this approach is not applicable in general. In contrast, introducing the virtual template enables our proposed method to provide such a point. Consider a feature point that resides in the undeformed area of the target image. The 3D coordinate of such a point on the reference template and on the deformed template is the same, i.e. $X^d = X^r$. Since X^r is known, this point can be used to constrain the deformed surface.

A similar issue exists in case of the reference template; and as a result, the calculated penetration depth can be a scaled version of the actual tool penetration depth. Following the same approach as before, the reference template should pass through a certain point on the object's surface. Let's take the reference image to be the one in which the tool-tip is in touch with the object. Therefore, if the tool-tip coordinate is known, say X^t , the coordinate of the point of touch is also

available, i.e. $X^r = X^t$ and the ambiguity is resolved. The coordinate of the tool tip is available in MIS in two ways; either from the forward kinematic of the robot (in RMIS) or from the image (considering cylindrical shape of the tools).

D. Force Estimation

It is worth expressing the relationship between the tool-tip displacement and its penetration depth. When a tool interacts with an organ, three different transformations can happen: rotation, translation and deformation. While the penetration depth is only related to the deformation transformation, the tool-tip displacement is a combination of all three.

A quantitative study on biomechanical characteristics of seven abdominal organs was reported in [3]. Several phenomenological models were fitted to the collected experimental data and it was shown that the function that best described the relationship between the applied stress, σ , and the strain, ϵ , was:

$$\sigma = \beta(e^{\alpha\epsilon^2} - 1) + \gamma\epsilon. \quad (6)$$

where α , β and γ are the model parameters. If the applied force F interacts with the object over area A and causes a penetration as deep as Δ , we have $\sigma = F/A$ and $\epsilon = \Delta/L$, where L is the thickness of the object in the direction of the applied force. Therefore, the relationship between the applied force and the penetration depth would be available.



Figure 3: The experimental setup. a) Force/torque sensor, b) PUMA robot, equipped with a laparoscopic surgical tool.

VI. EXPERIMENTAL RESULTS

To validate the applicability of the proposed approach, we setup an experiment. We used a PUMA robot to poke a lamb liver. An ATI Gamma force sensor was installed at the tip of the robot's hand and a laparoscopic tool was attached to the force sensor. The camera was installed on the experiment table (it was fixed to prevent camera motion ambiguity). The tool started in a position (almost 3 cm) above the organ and

moved on a straight line towards the liver, with a constant velocity. It stopped after traveling 45 mm on the line. The straight line was aligned with the z-axis of the force cell. Figure 3 demonstrates our experimental setup and Figure 4 shows three snapshots from the tool-organ interaction video.

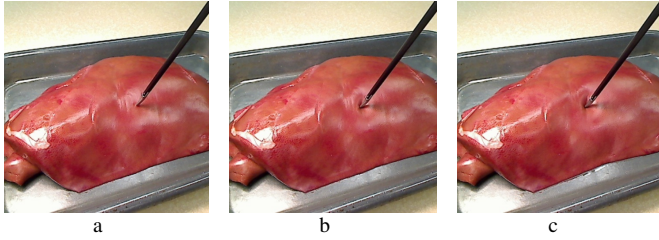


Figure 4: Three snapshots from the tool-organ interaction video, a) at the beginning of the deformation, when tool is in touch with the organ, b) at time when tool is half-way through, c) at the end of the motion.

Figure 5 shows the graph of a typical measured force signal. The signal indicates the force that is exerted to the liver by the tool. At time $t=2s$, the robot started to move, introducing some noises to the signal. Around time $t=7s$, the tool was in touch with the organ and the force interaction started. Finally around time $t=9s$, the robot stopped moving, the motion noise vanished and the force signals started to settle. Note that, although the motion was in z-axis direction, the force had components in x and y directions. This was due to the displacement of the liver (rolling) from its original pose. As it is appear in Figure 5, after the end of the robot's motion some force interaction existed that can be due to the organ damage (a mark remained on the organ surface after the tool removed). Since this behavior introduces error to our measurements, we only considered data collected before this behavior, e.g. the period of time between $t=7s$ and $t=9s$ in Figure 5.

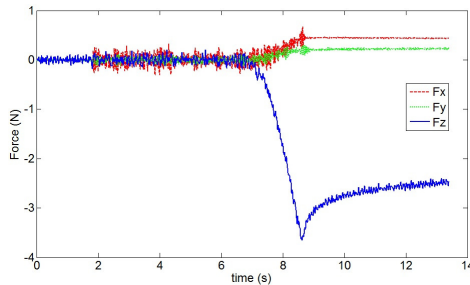


Figure 5: The force interaction between the tool and the organ.

To implement our method, we chose Non-uniform rational basis spline (NURBS) as our acceptable surface model [15]. To model a surface in NURBS, a grid of $m \times n$ control points, P_{ij} , is required. Each control point is associated with a rational base function, R_{ij} . The surface model is then described as:

$$X(s, t) = \sum_{i=1}^m \sum_{j=1}^n R_{ij}(s, t) P_{ij} \quad (7)$$

The next step in implementing the proposed method was feature detection and matching. As a well-known issue with the MIS cameras, the light source is placed next to the camera lens. Additionally, the objects (organs) are very close to the camera (around 3 inches away). This produces many saturated areas in the image, where we only see the white

patch of light reflections. Adding the wet nature of the environment inside the body to the situation, it is hardly possible for any standard feature matching algorithms to perform. As a result, we selected and matched the feature points manually for our experiment.

As discussed in IV.B, the number of required feature points is at least half of the parameters that we need to calculate. Due to the limited number of matched feature points in these images, we needed to reduce the number of shape variables. Here we used a grid of 5×5 control points, in which the central point contributed to the apex configuration, its surrounding eight points encoded the concavity of the deformed surface and the remaining 16 points encoded the deformation opening. Figure 6 depicts an example of the position of the control points and the resulting surface. To reduce the number of parameters even more, we added some constraints to the control points, e.g. the outer 16 control points resided on the reference template and the deformed surface should pass through these points.

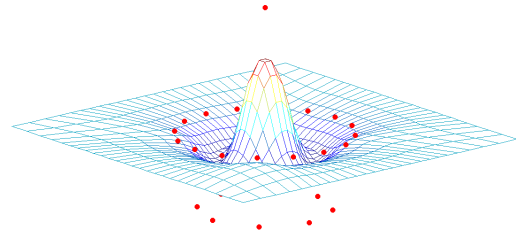


Figure 6: An example of a deformed surface, modeled in NURBS. The whole deformation structure is captured with only 25 points (red dots).

To address the scale ambiguity in the reference template, we obtained the exact coordinate of the tool-tip from the robot's forward kinematics. For scale ambiguity of the deformed template, we located an undeformed patch on the target image, among other methods, by calculating "optical flow" between the two images.

Figure 7 illustrates the results of running the algorithm for selected three frames of the video (shown in Figure 4). The reconstructed surface is shown below each image. Note that, the depth, orientation and position of the deformed patch changes, as in the images. The penetration depth would be the distance between the deformation apex and the center of the deformation opening.

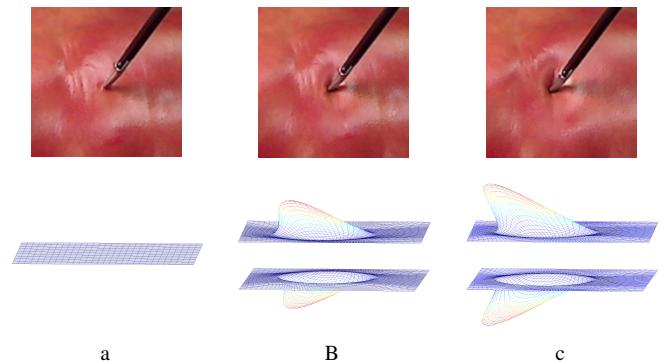


Figure 7: Three images and their reconstructed deformed surfaces, a) the reference image and the virtual template, b) image of the deformed liver (at the time when the tool is half-way through) and its deformed template from a top-side and a bottom-side views, 3) image of maximum deformation (at the end of the motion) and its template from a top/bottom-side views.

The calculated penetration depths for 10 selected frames from the video are shown in Figure 8, along with the estimated and measured force signals. As depicted in Figure 8.a, the estimated penetration depths closely follow a line. This is in agreement with the constant velocity motion of the robot on a straight line. In this particular experiment, the tool reaches the maximum penetration depth, estimated 13 mm, within 1.2 seconds.

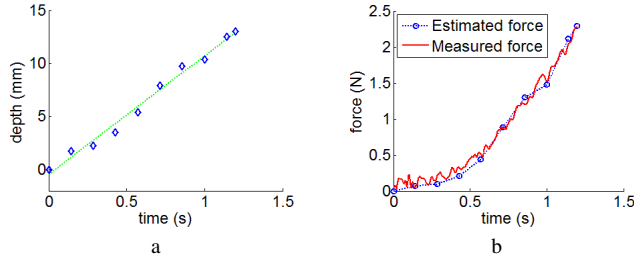


Figure 8: The experimental results, a) calculated penetration depth during the experiment (diamond) along with the linear regression (dashed line), b) estimated force signals (dashed line w/ circle) and measured one (solid).

The force estimation was carried out based on (6). To obtain values of α , β and γ for a lamb liver, we followed the same steps as reported in [3] (that study is based on porcine organs). The average values of the parameters for a lamb liver are $\alpha = 2.03$, $\beta = 4.99 \text{ MPa}$ and $\gamma = 0.35 \text{ MPa}$. The tool-tip area was $A = 3 \text{ mm}^2$ and the liver thickness was $L = 5 \text{ cm}$, approximately. The estimated force is shown in Figure 8.b, together with the magnitude of the measured force. As illustrated in this figure, the estimation error is considerably small. The maximum absolute error was less than $\text{Error}_{\text{max}} = 0.12 \text{ N}$ and the root mean square error (RMSE) was $\text{Error}_{\text{RMS}} = 0.07 \text{ N}$.

VII. CONCLUSION

The problem of lack of haptic information in minimally invasive surgical operations is considered in this paper. The proposed method estimates the tool-organ force interaction from monocular camera images with a high accuracy. The method consists of a deformation reconstruction algorithm and a force estimation step. The performance of the method was proved for an ex corpus test of a lamb liver, where the RMS of the force estimation error (RMSE) was 0.07 N.

In contrast to the template-based deformation reconstruction method, our virtual-template-based approach does not require the exact template of the organ; only smoothness assumption is sufficient. Moreover, unlike NRSFM methods, the proposed algorithm does not require the whole sequence of images in a video; a reference image and a target image are sufficient. Like many other machine-vision methods, our proposed method relies on the correctness of the feature matching process. However, as it was presented in previous section, by meaningfully constraining the surface model, our algorithm efficiently works with very few feature points.

Although we included force estimation as a part of our method, measuring the required parameters of human living organs (in vivo tests) are both impractical and patient dependent. We employed the force estimation mainly to

evaluate the performance of the proposed method, in absence of a ground truth measure for deformation reconstruction algorithm. In fact, in our ongoing research, the next step is to study on a direct relation between the deformation structure and its associated damage. Numerous skillful surgeons successfully operate MIS every day without perceiving any force information, which proves the efficiency and effectiveness of such a mapping. Our proposed algorithm integrated with this mapping will provide a cost effective solution for operation rooms with monocular vision.

REFERENCES

- [1] Xin H, Zelek JS, Carnahan H., "Laparoscopic surgery, perceptual limitations and force: A review", Conf. on Biomed. Comp.; Kingston, Ontario, Canada. 2006. No. 144.
- [2] Okamura, A.M., "Haptic feedback in robot-assisted minimally invasive surgery," *Curr Opin in Urology*, 19(1):102-7, 2009.
- [3] Rosen, J., Brown, J.D., De, S., Sinanan, M., and Hannaford, B., "Biomechanical properties of abdominal organs in vivo and postmortem under compression loads," *Journal of Biomech Eng Trans. ASME*, 130(2): 1-17 (2008)
- [4] Stoyanov, Danail, Ara Darzi, and Guang Zhong Yang. "Dense 3D depth recovery for soft tissue deformation during robotically assisted laparoscopic surgery." In *Medical Image Computing and Computer-Assisted Intervention—MICCAI 2004*, pp. 41-48.
- [5] Chao, L., Pinkall, U., Sanan, P., and Schroder, P., "A simple geometric model for elastic deformations", *ACM Trans. Graph.*, 29(4):38:1-38:6, 2010.
- [6] M. Salzmann, R. Urtasun, and P. Fua. "Local deformation models for monocular 3D shaperecovery," in *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1-8, 2008.
- [7] M. Eck and H. Hoppe. "Automatic Reconstruction of B-Spline Surfaces of Arbitrary Topological Type". In *ACM SIGGRAPH*, pages 325–334, 1996.
- [8] M. Perriollat, R. Hartley, and A. Bartoli. "Monocular Template-Based Reconstruction of Inextensible Surfaces". *International Journal of Computer Vision*, 2010.
- [9] M. Salzmann, R. Hartley, and P. Fua. "Convex Optimization for Deformable Surface 3D Tracking". In *International Conference on Computer Vision*, October 2007.
- [10] J. Zhu, S. Hoi, C. Steven, Z. Xu, and M.R. Lyu. "An Effective Approach to 3D Deformable Surface Tracking". In *European Conference on Computer Vision*, pages 766–779, 2008.
- [11] C. Tomasi and T. Kanade. "Shape and Motion from Image Streams Under Orthography: A Factorization Method". *International Journal of Computer Vision*, 9(2):137–154, 1992.
- [12] R. Hartley and R. Vidal. "Perspective Nonrigid Shape and Motion Recovery". In *European Conference on Computer Vision*, October 2008.
- [13] Salzmann, M., & Fua, P. "Deformable surface 3d reconstruction from monocular images". *Synthesis Lectures on Computer Vision*, 2(1), 1-113, 2010.
- [14] J. Fayad, A. Del Bue, L. Agapito, and P. M. Q. Aguiar. "Non-Rigid Structure from Motion Using Quadratic Deformation Models". In *British Machine Vision Conference*, 2009.
- [15] Les Piegl "On NURBS: A Survey", Jan 01, 1991, *IEEE Computer Graphics and Applications*, Vol. 11, No. 1, pp. 55 - 71