EDGES BASED SURGICAL INSTRUMENT TRACKING IN LAPAROSCOPIC IMAGES

G. M. Miranda*, P. H. S. Perruci*, R. F. Vergara* and M. C. Bernardes*

* Laboratório de Automação e Robótica (LARA), Universidade de Brasília, Brazil
e-mail: gabrielmirandatt@gmail.com, pedro.perruci@gmail.com

Abstract: The replacement of arduous tasks by machines has been growing in last years, promoting convenience and ease in different areas. Robotics in medicine sets one of these contexts, particularly in minimally invasive surgery, the scope of our work. In this paper, we propose a method for tracking surgical instruments in laparoscopy, shipped in a robotic endoscope holder and used in aid for surgeons in SUS, the Brazilian health system. The approach is based on the extraction of edges from endoscope images in a non-stereo setup and without previous knowledge of patient's positioning. The algorithm was validated with images from a database of real endoscopic videos, achieving a mean error of 15.53 pixels for high-quality images and 17.82 pixels for low-quality according to our metrics. For benchmarking, we considered the mean fps, that in our system reached a value of 24.62 fps. The obtained errors are compatible with an application for endoscope automatic positioning from images. Keywords: tracking, laparoscopic, vision, edges-based

Introduction

Laparoscopic surgery is one of most performed image-guided minimally invasive procedures. Generally, it is performed by two surgeons: the main physician conducts the procedure by the means of long and thin instruments inserted on the patient’s abdominal cavity, while an assisting surgeon is needed for holding the endoscope lenses inside the patient during the entire procedure. For both, Operation Room (OR)'s fatigue has been reported as a serious ergonomics concern [1].

Medical robotic endoscope holders have been developed to replace the assistant surgeon in the OR, leaving the camera's control to the main surgeon. Systems such Aesop [2] and Vicky [3] endoscope holders offer the surgeon full control over the visualization of the surgical operation and have an increasing interest of the medical society. However, due to the high costs of importation of any of these products for use in Brazilian's health system (SUS), University of Brasília's project CLARA aims to develop a national alternative for the technology.

Even though robotic endoscope holders have been designed to receive commands by ergonomically comfortable means, the surgeon still needs to deviate his attention of the procedure to move the device, which might compromise the procedure. The present work proposes a new method for visual tracking of surgical instruments that could be used to automatically command CLARA's robotic endoscope holder. Our main intention on tracking the tip of the laparoscopic instrument is to autonomously orient the movement of the endoscope holder in order to center the images at the surgical tool whenever requested by the surgeon.

Related methods from the state of the art report some variety of approaches. The simplest and most effective approach is tracking a colored marker, added to the tip of the instrument [4]. Some strategies were even able to acquire 3D positioning from an appropriate set of markers [5]. Although their results seem promising, the need of biocompatible markers restrains the solution to custom made marked instruments. A wider range approach uses a classifier, similarly to face tracking algorithms applied to surgical instruments [6]. It has been reported near real time computer efficiency [7], although the method involves intensive training of carefully chosen datasets. The most generic and direct approach found in the literature is to track the surgical instruments by the edges found on image. Though it involves high computational cost processing, this solution is able to extract the tool tip position with good precision. More ambitious approaches use the insertion point's position of each instrument to filter edges and estimate the 3D position of the tip near real time [8].

This paper presents a novel visual tracking method for surgical instruments. It comprises an edges based approach for 2D pose estimation of the instrument's tip. The main contribution of this research is to develop a low cost surgical instrument tracker, robust to occlusion and color changes. The method requires no prior knowledge about the surgical environment and the main restraint is the need of appropriate contrast between the instrument's axis and the background of the video frame. Although the algorithm was designed to be embedded on CLARA's robotic endoscope holder, the solution is expected to develop properly in different types of minimally invasive operations.

Materials and methods

The proposed approach for tracking surgical instruments in laparoscopic videos is based on the recognition of the tool geometric characteristics on the image. The tool shaft can be represented in the image by two lines, almost parallel and close to each other. These conditions are sufficient to filter probable lines from those present in the image, locate the extremity of the shaft and estimate the instrument's tip position. Known geometrical constraints concerning the tool's shaft have been reported by [8] and [9] and have shown precise and robust results for stereo camera images.

In this paper we propose a method based on these previous studies, but adapted to monocular endoscope captures with no prior knowledge about the surgical environment. These are the conditions commonly found on SUS. As a consequence, we developed a robust method to estimate the 2D pose of a single instrument tip.

Our implementation can be summarized in five stages: preprocessing, edges extraction, lines extraction, tools shaft extraction and locate instrument's tip. The process workflow is represented in Figure 1 and the method is summarized in Algorithm 1. Each stage will be briefly explained on the following subsections.

Preprocessing – The first part of our implementation consists on detaching the surgical tools from the image's background. Also, luminous variations as shadows must be smoothened. Therefore, it is performed a histogram equalization on each of the image channels. Equalized histogram increases the global contrast of the image, allowing a better visualization of the instrument.

Edges extraction – Then, a color threshold is applied, clustering similar characteristics of the environment in order to differ the black surgical tools from the red background on the patient's body. In the sequence, Canny edges detection algorithm is used. It consists in extracting image's gradients and select best edge points candidates by applying a double threshold on their gradient values. To avoid the effect of poorly illuminated regions in the corners of the frame, we consider an ellipsoid Region Of Interest (ROI) that ignores those darker regions.

Lines extraction – We focus on searching for the two lines that best fit the instrument's shape. Upon the edges image, Hough Transform [10] is called to extract lines formed by them. It consists in a voting operation in which every possible lines passing through each pair of edge points is computed and stored in the Hough Space — a Cartesian plan, in function of line's polar coordinates (ρ,θ). As a result, it's obtained a vector of lines coordinates, ordained by their votes.

Tool shaft extraction – In order to identify the instrument's body among all lines found on image, an outer loop was designed to filter the lines upon their relevance and the expected geometrical constraints. Because Hough Lines already outputs lines ordered by votes, it is assumed that the first line of the vector belong to the instrument's axis. So, an inner loop iterates through the vector searching for its pair. The second line must be close to the first line in both their polar parameters ϱ and θ, meaning that $\theta min < \Delta\theta < \theta max$ and $\rho min < \Delta\rho < \rho max$. When a valid pair is found, the inner loop — for the second line — is satisfied and an instrument's body is found. Then, the outer loop goes on, searching for instrument's lines which fulfill the geometrical restraints until the vector of lines is done. By the end, it's obtained a vector of possible instrument's bodies on the image.

Tool tip location – The primary objective of this step is to locate the end of the tool's shaft in each of the instrument's bodies. A loop goes through the vector of bodies, analyzing each pair of lines. It's assumed that both of the axis ending points are represented as edges returned by the Edges extraction step. So, a possible ending point is each point of the edges image which belongs to the line. A step for finding ending point candidates is performed.

In order to locate the actual, or the best, ending point Po, it has to have neighbor points at one side, though none in the other. A tolerance γ for this computation is used. If the point Po can be written as

$$Po = Po.x + Po.y \, ,$$

then γ is

$$\gamma = \gamma.x + \gamma.y.$$

In order to estimate the tip's location through both body's lines, we need to correspond Po to its pair, Po'. These points must be positioned close to each other, respecting a second restraint, given by

$$\gamma - Pt < \Delta Pt < Pt + \gamma.$$

When a successful correspondence occurs, the instrument's tip estimation is expressed as the arithmetic mean of the corresponded points.

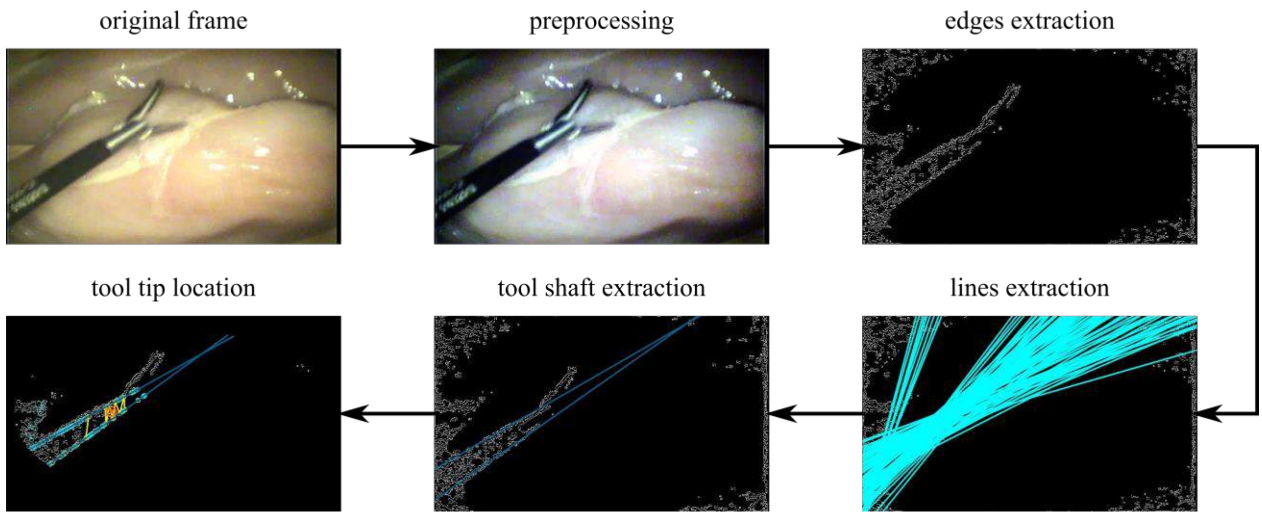

Figure 1: Workflow of the proposed algorithm: the original laparoscopic video frame goes through preprocessing, extraction of edges, lines and tool shaft, and location of tool tip, which is returned as output of the implementation.

Algorithm 1 Pseudocode for surgical tool detection

```
procedure LOCATE TOOLS TIP
    Begin
    image ← videoFrame
    image ← PREPROCESSING(image)
    image ← EDGESEXTRACTION(image)
    vector of lines ← EXTRACTLINES(image)
    loop each line in vector of lines
        first line ← Most Relevant Line(line)
        loop each line' in vector of lines
            second line ← FulfillConstraints(first line, line')
        end loop
    end loop
    if there is( second line ) then
        Tool Shaft ← (first line, second line)
        loop each of shaft's lines
            vector of points ← ExtractEdgePoints(Tool Shaft)
            ending points ← FindEndPoints(vector of points)
        end loop
        TipEstimation ← CorrespondPoints( ending points )
        Tip ← Arithimetic Mean(TipEstimation)
    end if
    End
end procedure
```

Results

The presented solutions were developed in C++ language, based on OpenCV's libraries. The images were obtained from monocular recording of in vivo MIS from the web database “Hamlyn Centre Laparoscopic / Endoscopic Video Datasets”[1] and from the Dr. R.K. Mishra database [2]. To assure the robustness of our solution, we tested it to a variety of images from different surgical operations. Each of them shall represent a different set of illumination and surgical tool's color — determinant characteristics of an image for computer vision. In addition, ex-vivo simulations were recorded in laboratory with an endoscopic camera.

For evaluation, were selected a set of 573 frames of five different videos of our dataset. On each of them, a possible instrument shaft was detected. These frames were distributed in 304 high quality images and 269 low quality images and, for each of them, the ground truth (GT) was manually elaborated. The region considered as instrument’s tip consists in the entire metallic region in top of the instrument's body.

Each set of results is expressed in Table 1. Values considered for analysis were the absolute difference between the estimated tip position and the ideal position (μ) — in pixels — their standard deviation (σ) and a hit parameter. When the instrument's tip estimation error is less than 20 pixels, a hit is counted. It's valuable to notice that, when choosing the ideal position, the shortest distance between the real value and the metallic region of the tip in the image was considered.

Regarding the evaluation of our algorithm computational cost, we measured the average frame rate used to process the selected set of frames used for validation. The computer used for benchmarking was an Intel® CoreTM i3-2310M CPU @ 2.10GHz × 4 processor with 4 GB of RAM, which obtained a frame ratio of 24.62 fps.

Table 1: Tracking results obtained for method validation. The estimated pose was compared to a manually selected GT. When their difference is under a 20-pixel tolerance, a hit is computed.

Images	Mean error	Deviation	Hits ratio
High Quality	15.53 px	22.65 px	88.88%
Low quality	17.82 px	26.26 px	86.62%

Discussion

For both high and low quality image groups, the algorithm has shown a valuable precision rate. Although it couldn’t match [9]’s accuracy, the consistency between image groups is notable. This robustness is essential for general camera conditions, which can be found on SUS.

The average frame rate of 24.62 fps represents a computationally efficient solution, capable of running near real time. It means approximately a 9% advantage when compared to [9]’s 26.98 fps. Even though experimental setups differ and research objectives were more complex, their measurements where performed in a much more powerful computer. So, results obtained in this paper indicate an efficient solution for tracking 2D pose estimation.

However, the tool tracking algorithm was evaluated in restrained conditions and can find limitations when applied in other contexts. In our experiments, being the instrument shaft present or not, the algorithm would find a tip. However, this is not always true in a laparoscopic procedure, when the instrument tip may not be present at all or may be occluded. As a consequence, our implementation can present false positives.

Conclusion

The proposed method for 2D pose estimation of surgical tools has shown promising results. To the best of the authors knowledge, it is one of the few solutions that propose locating surgery tools in a monocular setup. Its main contribution relies on the method’s computational efficiency, which is faster if compared to stereo tracking methods. It has also shown a high precision rate for both high and low quality images.

The presented tool tracking algorithm was designed to be embedded in CLARA’s robotic endoscope holder. Hence, the initial requirement that it be fast and efficient enough to perform near real time tracking was accomplished. Also, it has shown considerable precision when finding the instrument’s tip in the proposed conditions.

For future developments, we intend to evaluate the solution in different conditions and its performance on a full video scenario. We also intend to replace the default

[1] http://hamlyn.doc.ic.ac.uk/vision/

[2] http://laptube.net/

Hough Lines method by its Probabilistic version with the purpose of decreasing, even more, the overall computational time. Using Hough's Probabilistic version may even increase the finding tip accuracy.

In order to expand the scope of our method, we may also investigate alternatives to handle printed characters usually present in the instrument's bodies.

Acknowledgements

This project is funded by the Brazilian Ministry of Health (Termo de Cooperação nº 121/2013, Processo 25000.169843/2013-83).

References

[1] A. N. Supe, G. V. Kulkarni, and P. A. Supe, "A computational approach to edge detection," J. Minim Access Surg., vol. 6, no. 2, pp. 31–36, 2010.

[2] P. Ballester, Y. Jain, K. Haylett, and R. F. McCloy, "Comparison of task performance of robotic camera holders endoassist and aesop," in International Congress Series, vol. 1230, pp. 1100–1103, Elsevier, 2001.

[3] J.-A. Long, P. Cinquin, J. Troccaz, S. Voros, P. Berkelman, J.-L. Descotes, C. Letoublon, and J.-J. Rambeaud, "Development of miniaturized light endoscope-holder robot for laparoscopic surgery," Journal of Endourology, vol. 21, no. 8, pp. 911–914, 2007.

[4] G.-Q. Wei, K. Arbter, and G. Hirzinger, "Realtime visual servoing for laparoscopic surgery. controlling robot motion with color image segmentation," Engineering in Medicine and Biology Magazine, IEEE, vol. 16, no. 1, pp. 40–45, 1997.

[5] L. Bouarfa, O. Akman, A. Schneider, P. P. Jonker, and J. Dankelman, "In-vivo real-time tracking of surgical instruments in endoscopic video," Minimally Invasive Therapy & Allied Technologies, vol. 21, no. 3, pp. 129–134, 2012.

[6] R. Sznitman, K. Ali, R. Richa, R. H. Taylor, G. D. Hager, and P. Fua, "Data-driven visual tracking in retinal microsurgery," in Medical Image Computing and Computer-Assisted Intervention–MICCAI 2012, pp. 568–575, Springer, 2012.

[7] M. Allan, S. Thompson, M. J. Clarkson, S. Ourselin, D. J. Hawkes, J. Kelly, and D. Stoyanov, "2d-3d pose tracking of rigid instruments in minimally invasive surgery," in Information Processing in Computer-Assisted Interventions, pp. 1–10, Springer, 2014.

[8] S. Voros, J.-a. Long, and P. Cinquin, "Automatic Detection of Instruments in Laparoscopic Images: A First Step Towards High-level Command of Robotic Endoscopic Holders," The International Journal of Robotics Research, vol. 26, no. 11-12, pp. 1173–1190, 2007.

[9] R. L. Dockter II, "A Fast , Low-Cost , Computer Vision-Based Approach For Tracking Surgical Tools," IROS Conference (IEEE), p. 198, 2013.