Visual shape and position sensing algorithm for a continuum robot

R A Manakov, D Y Kolpashchikov, V V Danilov, N V Laptev, I P Skirnevskiy and O M Gerget

Medical Devices Design Laboratory, Tomsk Polytechnic University, Tomsk, Russia

E-mail: ram290495@gmail.com

Abstract. Continuum robots represent an actively developing and fast-growing technology in robotics. To successfully implement control and path planning of continuum robots it is important to develop an accurate three-dimensional shape and position sensing algorithm. In this paper, we propose an algorithm for the three-dimensional reconstruction of the continuum robot shape. The algorithm is performed during several steps. Initially, images from two cameras are processed by applying pre-processing and segmentation techniques. Then, the gradient descent method is applied to compare two-dimensional skeleton points of both masks. Having compared these points, it finds a skeleton of the robot in a three-dimensional form. Additionally, the proposed algorithm is able to define key points using the distance from the robot base along the center line. The latter allows controlling the position of points of interest defined by a user. As a result, the developed algorithm achieved a relatively high level of accuracy and speed.

1. Introduction

Today the continuum robot technologies are actively developing. These robots obtain a higher dexterity compared to the traditional rigid robots. In this regard, they are capable of operating in complex workspaces. Due to this advantage, continuum robots are widely used in medicine for minimally invasive procedures [1]. For instance, robotic catheters are used in endovascular surgery as the main tools for the delivery of instruments and stents. Other fields of application of continuum robots are non-destructive testing and repair [2–5].

In order to successfully implement a closed-loop control and path planning of continuum robots, it is important to develop an accurate three-dimensional shape sensing algorithm. Due to a flexible robot structure, this task is quite challenging and complex. Currently there are several techniques for 3D shape sensing and some of them are fiber-optic sensor-based algorithms (FOS), electromagnetic-tracking (EM), and image-based methods [6].

Image-based methods are less complex, cheaper, and more accessible in comparison with FOS and EM techniques as they do not require any modification of the robot construction and/or sensors. It allows applying them using simple equipment such as webcams. Image-based methods are able to measure the shape of a robot directly and reconstruct it in 3D without additional kinematic modeling.

In connection with the foregoing, we propose an image-based algorithm which is able to estimate a continuum robot shape and its tip position inexpensively. In the algorithm's workflow, the gradient descent method is used for pairwise comparison of the skeleton points obtained by two cameras. Once this comparison is performed, it finds a skeleton of the robot in a three-dimensional form. Additionally, the proposed algorithm is able to define key points using the distance from the robot base along the center line. This algorithm feature allows controlling the position of the points defined by a user.

2. Methods

The workflow of the proposed continuum robot shape sensing algorithm is represented by several steps shown in Figure 1. During the first step, the algorithm simultaneously acquires images using two cameras. At least two cameras are creating a stereoscopic effect and allow us to receive information about the robot's spatial position.

In the second step, acquired images are being segmented and the algorithm recognizes the robot on them. To simplify the image processing operations and improve the detection accuracy, the image with the robot has a monotonous background. The process of image acquisition is carried out with uniform illumination of the scene and adjusted parameters of the cameras (brightness, contrast, saturation, exposure, and clarity). With these adjusted parameters the robot border is not blurred and there are no glare and other undesirable effects. It should also be noticed that the non-linear distortion has been removed from each image. The latter allows reducing the triangulation error. During the pre-processing step the algorithm performs such morphological operations as area opening and morphological closing. Area opening helped us to remove unnecessary small objects and noise, while morphological closing fills the gaps and blurred the edges of the masks. Initial segmentation is performed using RGB thresholding, calculated as follows:

$$\sqrt{(R - R_0)^2 + (G - G_0)^2 + (B - B_0)^2} > T \qquad (1)$$

where R, G and B are the current pixel colors; R_0, G_0 and B_0 are background colors, T is the threshold level.

Once the binary masks are obtained, the algorithm skeletonizes them using distance transform and then computes the bounding boxes. Image skeletonization is required for finding an axial line of the continuum robot on each image. In order not to process the whole image in the following steps, the algorithm defines bounding boxes of the region of interest.
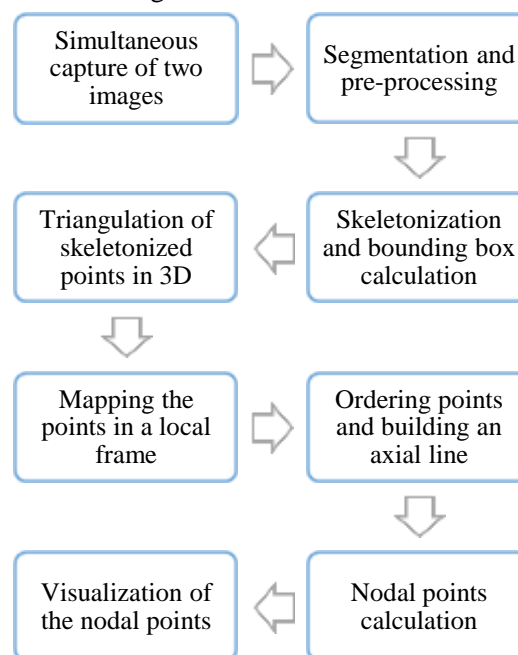

Figure 1. Steps of the proposed algorithm.

Pre-processing and segmentation operations are followed by the triangulation. For the triangulation of the points in three-dimensional space, we implemented the gradient descent method based on the triangulation function. This implementation allows the algorithm to define the pairs of corresponding points on both skeletons. The purpose of an application of triangulation is the error minimization between two rays (Figure 2). The first ray is fixed, and the direction of the second is chosen in such a way that the error of their triangulation is the lowest. The triangulation analyzes the current and the

neighboring pixels of the skeleton and iteratively moves along it until the objective function minimum is reached. The initial approximation of the method is the result of the minimization of the previous pair of rays.
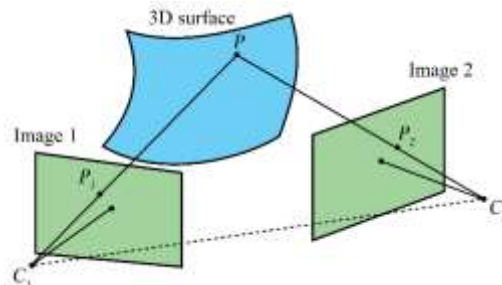

Figure 2. Minimization of the error between two rays.

This implementation allows the algorithm to minimize the number of iterations and speed up the triangulation process.

The described method finds the correspondence between the points of the first and second skeletons. Thus, we can find the pairs of points on two skeletons, triangulate them and receive a set of points corresponding to the robot skeleton in three-dimensional space. In order to obtain the maximum possible number of point pairs, the algorithm is applied to all the pixels of the skeleton on both masks. On the one hand, this may be redundant. On the other hand, such an approach makes it possible to obtain additional points for non-trivial forms of the robot.

For the convenience of further work with the obtained set of points in three-dimensional space, it has to be mapped into the local coordinate system of the robot. In this connection, we use an affine transformation matrix. In the local coordinate system, the robot is located at the frame origin, and its longitudinal axis is directed along the Z-axis. All the further calculations are carried out in this coordinate system.

For the obtained set of points, it is necessary to define the correct order and remove duplicate points. In order to achieve that, the Euclidean metric is used. The process of sorting starts from the robot base. In the beginning, the algorithm takes the point with the smallest Z-coordinate. The next point will be the closest to the selected point. This process is repeated iteratively until the exact order is found for all sets of points. As a result, we obtain a sorted sequence describing the robot center line.

During the last step, the algorithm searches for the robot points defined by a user. The idea behind this step is to set the distance for each point from the robot base along the axial line.
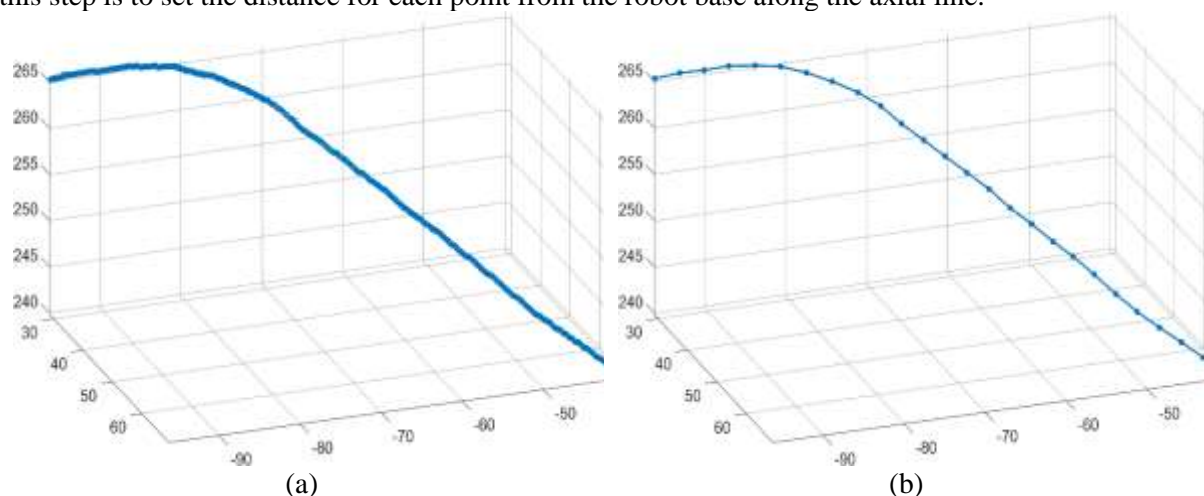

(a)   (b)

Figure 3. 3D skeleton and nodal points of continuum robot: a) Triangulated points of the 3D skeleton. b) Calculated nodal points.

Thus, the search for these points is reduced to the integration of the previously obtained sequence along the length of the center line. The beginning of the integration is a point on the center line with some coordinate Z_0, determined by the robot base. It is worth noticing that the linear interpolation is actively used for improving positioning accuracy. As a result of integration, the exact coordinates of the nodal points representing an output of the detection algorithm are defined. If necessary, these points can be converted back to the stereo camera's frame using the affine transformation calibration matrix. Founded axial line and corresponding nodal points are reflected in Figure 3.

3. Testing and Results

The proposed algorithm was tested using a two-section continuum robot. Each section is 40 mm length and has two degrees of freedom. Additionally, the laboratory test bench shown in Figure 4 has five degrees of freedom including the linear movement. As shown above, the laboratory test bench has a monotonous background and two cameras. Image acquisition is carried out with good illumination and well-adjusted parameters.

In order to acquire images in real-time, we used two Logitech C922 Pro Stream cameras at the resolution of 1920x1080. As for the software, the Computer Vision Toolbox of MATLAB R2019a was used for calibration and work with the cameras. Testing was conducted on a computer equipped with an Intel Core i7 CPU and 16 Gb RAM. All the results were obtained at a resolution of 1920x1080. The average error of the triangulation made up 0.215 mm. The final accumulated integration error did not exceed 0.07 mm for every 10 mm of the robot length. Having tested the algorithm's execution time, the average detection time made up approximately 0.7 seconds. Two examples of the source, segmented, skeletonized, and detected images are presented in Figure 5.
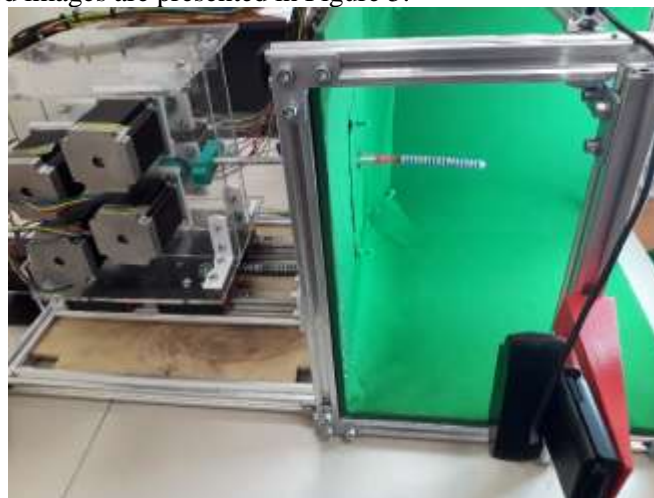

Figure 4. Laboratory test bench for continuum robot shape sensing.

The proposed algorithm is compared with two other well-known approaches. One of them is represented by the algorithm of D. Camarillo et al. [7], another is reflected in the paper of A. Vandini et al. [8]. D. Camarillo et al. use the algorithm where images from three cameras are pre-processed by the RGB thresholding obtaining binary masks. Then the voxel carving technique is applied. According to this technique, the volume of interest is divided into a discrete three-dimensional grid of volume elements (voxels). Each voxel is back-projected onto the source images and is classified as part of the 3D shape if it lies within the masks. In related research, A. Vandini et al. conduct experiments with a planar continuum robot attached to the rigid robot. In this study, a single-camera and a naive Bayesian classifier are used to extract the robot center line. The three-dimensional shape is reconstructed from the central line, position, and orientation of the rigid robot. An accuracy comparison of the proposed algorithm with the above mentioned image-based algorithms is reflected in table 1.

| | (a) Source image | (b) Segmentation | (c) Skeletonization | (d) Detection |
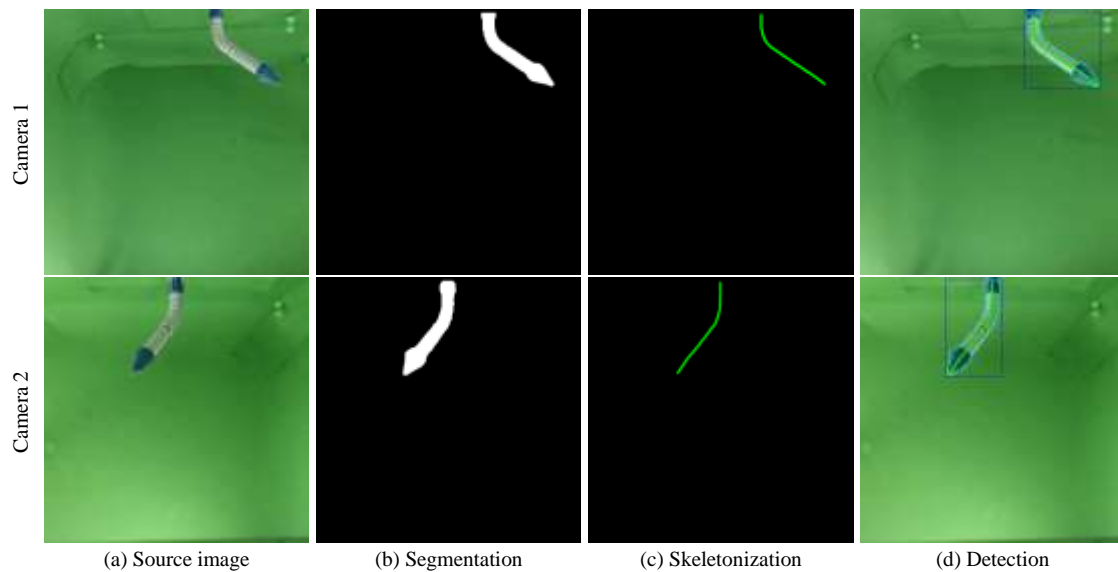
Figure 5. Examples of the algorithm outputs.

Table 1. Comparative Evaluation of the Algorithms.

Algorithm	Error	Reference
Proposed algorithm	0.07 mm per 10 mm of the robot length	–
Vision-based 3D shape sensing of flexible manipulators	8 mm for 160 mm robot	[7]
Vision-based motion control of a flexible robot for surgical applications	1.84 mm	[8]

4. Conclusion

In this paper, we presented the algorithm for the shape and position sensing of continuum robots. The proposed algorithm uses the image-based approach with two stereo cameras. Having tested the algorithm, the accumulated integration error did not exceed 0.07 mm for every 10 mm of the robot length. The average execution time made up about 0.7 seconds. The presented results show that the algorithm is able to estimate the continuum robot shape and define its spatial position. The proposed detection algorithm can be easily implemented and does not require any special expensive hardware. The algorithm is not capable of working in real-time but this issue can be solved by its implementation using C++ and its computer vision libraries, for instance, OpenCV. The final accuracy can be improved by using a higher resolution of the cameras or by increasing the number of cameras.

Acknowledgments

References

[1]   Burgner-Kahrs J, Rucker D C and Choset H 2015 *IEEE Trans. Robot.* **31** 1261–80
[2]   Dong X, Axinte D, Palmer D, Cobos S, Raffles M, Rabani A and Kell J 2017 *Robot. Comput. Integr. Manuf.* **44** 218–29
[3]   Dong X, Palmer D, Axinte D and Kell J 2019 *J. Manuf. Process.* **38** 313–8
[4]   Buckingham R and Graham A 2012 *Ind. Robot An Int. J.* **39** 6–11
[5]   Mehling J S, Diftler M A, Chu M and Valvo M 2006 *First IEEE/RAS-EMBS Int. Conf. Biomed.*

14th International Forum on Strategic Technology (IFOST 2019)   IOP Publishing

IOP Conf. Series: Materials Science and Engineering   1019 (2021) 012066   doi:10.1088/1757-899X/1019/1/012066

Robot. Biomechatronics, 2006, BioRob 2006 **2006** 690–695

[6]   Shi C, Luo X, Qi P, Li T, Song S, Najdovski Z, Fukuda T and Ren H 2017 *IEEE Trans. Biomed. Eng.* **64** 1665–1678

[7]   Camarillo D B, Loewke K E, Carlson C R and Salisbury J K 2008 *Proceedings - IEEE International Conference on Robotics and Automation* 2940–2947

[8]   Vandini A, Salerno A, Payne C J and Yang G Z 2014 *Proceedings - IEEE International Conference on Robotics and Automation* 6205–6211