2020 13th International Conference on Developments in eSystems Engineering (DeSE) | 978-1-6654-2238-3/20/$31.00 ©2020 IEEE | DOI: 10.1109/DeSE51703.2020.9450784

Medical palpation autonomous robotic system modeling and simulation in ROS/Gazebo

Artur Shafikov, Tatyana Tsoy, Roman Lavrenov, Evgeni Magid
Laboratory of Intelligent Robotic Systems
Intelligent Robotics Department,
Institute for Information Technology and Intelligent Systems (ITIS)
Kazan Federal University, 35 Kremlevskya street,
Kazan, 420008, Russian Federation
artur.shafikov96@gmail.com, {tt, lavrenov, magid}@it.kfu.ru
kpfu.ru/robolab.html

Hongbing Li
Department of Instrument Science and Engineering,
Shanghai Jiao Tong University
China, Shanghai, Minhang, 200240
lihongbing@sjtu.edu.cn

Elena Maslak
Department of Pediatric Dentistry
Volgograd State Medical University
Volgograd, 400131, Russia
eemaslak@yandex.ru

Natalia Schiefermeier-Mach
Health University of Applied Sciences
6020 Innsbruck, Austria
natalia.schiefermeier-mach@fhg-tirol.ac.at

***Abstract*—Manual medical palpation task performed by a human doctor is characterized by complexity of multiple source data fusion and decision subjectivity, which significantly depends on a doctor experience and sensitivity. Therefore automated robotized approach could provide more reliable and independent results relative to a manual palpation. This paper presents a model and the Gazebo simulation of an autonomous robotic system that performs palpation of an arbitrary soft surface. The system uses a modified KUKA IIWA LBR manipulator model, equipped with a spherical indenter and a new force sensor plugin. In order to perform a palpation procedure, the system gathers data from a virtual Kinect camera and computes points for palpation based on an acquired point cloud. We introduce an implementation of soft bodies features with variable stiffness to the Gazebo simulator based on DART physics engine capabilities. The developed system was tested in the Gazebo simulator by simulating palpation of abdomen soft model with variable stiffness over its surface. The system successfully performed the palpation and was able to detect an area of high stiffness.**

***Index Terms*—autonomous palpation, robotic palpation, soft body simulation, robotics, coverage algorithm**

I. Introduction

Medical palpation is a physical examination technique that uses a sense of touch as a source of information for medical diagnosis [1]. Among factors assessed by palpation is a presence of guarding, rigidity, lumps, masses and other abnormalities [2] that might indicate a presence of diseases, including peritonitis, appendicitis and tumors, e.g., a kidney cancer, a pancreatic cancer, a fibroadenoma, a breast cancer etc [1]. Two major types of palpation are a superficial (light) palpation and a deep palpation. While the superficial palpation is used for a preliminary examination that involves noting abnormal masses and lumps, the deep palpation is used for further investigation of those masses and lumps and, if accessible, to assess internal organs of a palpated area [3]. A palpation depth varies from about one centimeter for the superficial palpation and to eight centimeters for the deep palpation [2].

A significant experience required for mastering the palpation skill [4] and a subjective nature of a palpation procedure are the main reasons of research activity in the area of palpation automating. Another motivation is a telemedicine that aims for patients who have difficulties in accessing medical services [5]. Existing research covers a wide range of autonomous palpation aspects including palpated surface exploration techniques [6], [7], abnormality assessment [8], and sensing techniques [9], [10]. However, to the best of our knowledge, autonomous palpation system modeling and simulation are not covered thoroughly in present researches. An additional difficulty is modelling a contact between a robot and a soft body in the Gazebo robotic simulator [11], [12].

This paper describes a model of autonomous robotic system based on a robotic manipulator that is capable to perform superficial palpation of an arbitrary surface based on data from a Kinect camera. We developed a virtual implementation of KUKA IIWA LBR manipulator [13], [14] based model in Gazebo simulator with soft body features implemented. DART physics engine capabilities were used for the soft body features. The developed system was tested by simulating palpation of abdomen soft model with variable stiffness over its surface. The system successfully performed palpation of the abdomen and was able to detect an area of high stiffness.

II. Simulation setup

A. Simulation environment

Simulation environment in the Gazebo (Fig. 1) contains a modified KUKA IIWA LBR manipulator model (Fig. 2) from

our previous work [15]. The manipulator is placed on a cubic supporting base that shifts the workspace in order to cover surgical table surface. A Kinect camera model with Robot Operating System (ROS) depth camera plugin is mounted on a tripod to enable palpated surface geometry data acquisition. A human abdomen model is placed on the table within the manipulator workspace and the Kinect camera range [16].

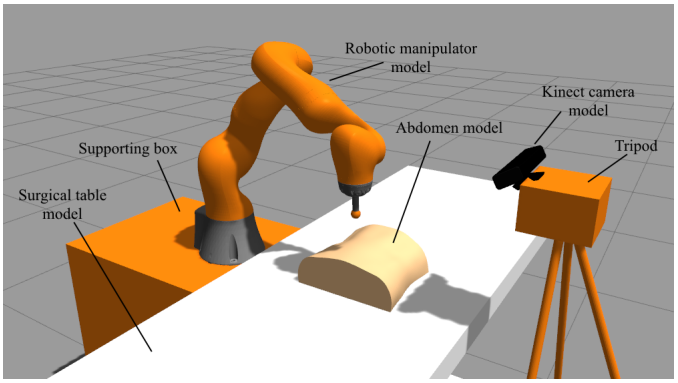

Fig. 1. A virtual environment in the Gazebo simulator.

B. Soft object simulation

Dynamic Animation and Robotics Toolkit (DART) physics engine [17] has a soft body functionality. It is supported by the Gazebo simulator that could be built using the DART and the Gazebo source files. However this feature is not implemented in the Gazebo simulator, including the Gazebo 9 version that was used in our research. To overcome this limitation, the original source code of the Gazebo was modified to allow the following functionality:

- Correct processing of the DART soft surface properties available in Simulation Description Format (SDF) model specification.
- Generating collision, inertia and mass of a soft body taking into account its rigid (bone) and soft (flesh/skin) parts.

Soft body feature of the DART engine is based on a mass-spring system that represents a soft surface as a set of point masses and elastic forces, applied on each point mass, are modeled as linear spring forces [18]. The relationship between a force and a deformation is described by Hooke's law:

$$F = kx \tag{1}$$

where F is a force arising from a contact with a point mass, x is a deformation of the point mass spring and k - is a stiffness of the point mass spring.

While there is a default DART method to create a simple shape soft body (e.g., a box, a sphere or a cylinder), no such method exists for an arbitrary mesh. Our implementation treats each vertex of a mesh as a point mass, thus the mesh structure defines point mass quantity and distribution. In order to get consistent data from a force sensor while probing a soft surface, it is essential to use a model with a large number of

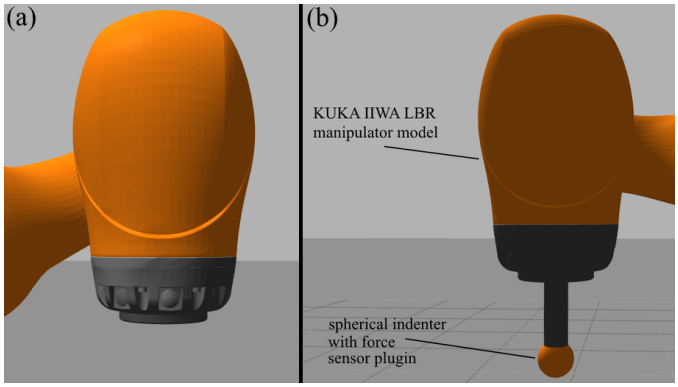

Fig. 2. (a) An original KUKA IIWA LBR manipulator model; (b) A modified model.

vertices. That results into a dense even distribution of point masses across the soft surface and should guarantee that an end-effector interacts with approximately the same number of vertices and edges during each surface contact.

To generate a rigid part of the soft body, scaling of the original model was used. The SDF format allows specifying only a mass ratio between a soft and a rigid body parts; this ratio is also used to compute a scaling factor thus specifying volume distribution between the soft and the rigid body parts. Properties of soft bodies are specified via SDF description of the model, and the following four parameters are available for customizing [19]:

- *Vertex stiffness* and *edge stiffness* (bone_attachment and stiffness tags in SDF) - define a magnitude of a normal force arising as a result of collision with a vertex and an edge of the soft mesh model respectively.
- *Damping* (damping tag in SDF) - defines a damping of a normal force arising as a result of a collision between a soft body and another body.
- *Flesh mass fraction* (flesh_mass_fraction tag in SDF) - defines a mass and a volume ratio between a soft (a flesh/skin) and a rigid (a bone) body parts.

To test a soft body implementation we developed a force sensor plugin that acquires force data directly from the DART physics engine. This plugin was added to a link (end-effector), which was used for testing soft surfaces by plunging itself into the surface and collecting normal force and position data during the contact. Both plots, a normal force in Fig. 3(a) and a position in Fig. 3(b), for the implemented soft surface demonstrated oscillations, which are common for elastic and soft surfaces. The plots for a rigid surface (Fig. 3(c) and (d)), which is a default surface type in the Gazebo simulator, demonstrated no oscillations. For testing we used the surface of 1.0085 meters height. When the end-effector contacts the rigid surface, it switches into an equilibrium state and stops on the rigid surface without any further motions or oscillations. For a soft surface after the end-effector contacts the soft surface at 0 seconds time (the spike in the normal force plot for the soft surface in Fig. 3(a)), the end-effector continues to

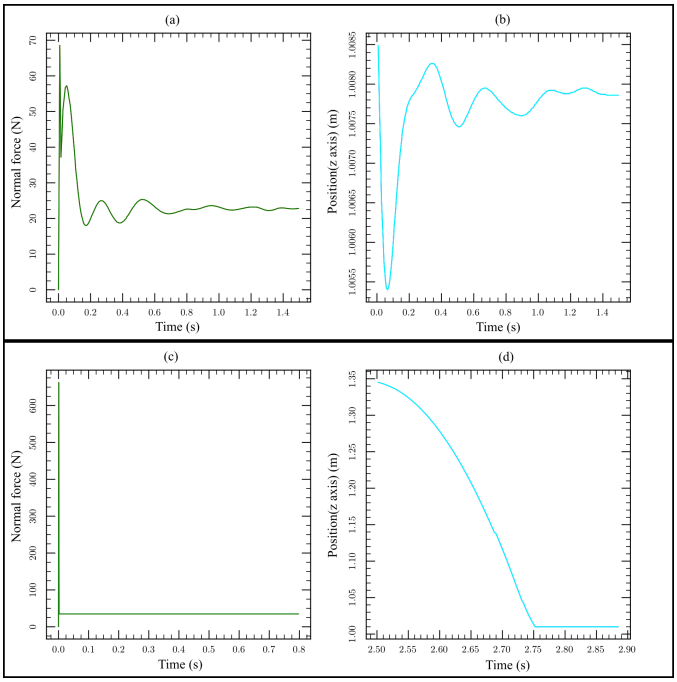

Fig. 3. (a) and (c): Normal force exerted on the end-effector by the soft and the rigid surfaces respectively; (b) and (d): Position (z axis) of the end-effector during the contact with the soft and the rigid surfaces respectively. For plots (a), (b) and (c) the contact occurs at 0 seconds time; for plot (d) the contact occurs at 2.75 seconds time.

plunge into the soft object (Fig. 3(b)). The position plot for the soft surface (Fig. 3(b)) demonstrates that the end-effector plunges into the surface by approximately 3 millimeters before being pushed out (up) by the normal force of the surface. As soon as oscillations are fully damped, the end-effector goes into the equilibrium state and stays plunged into the surface by approximately 1 millimeter.

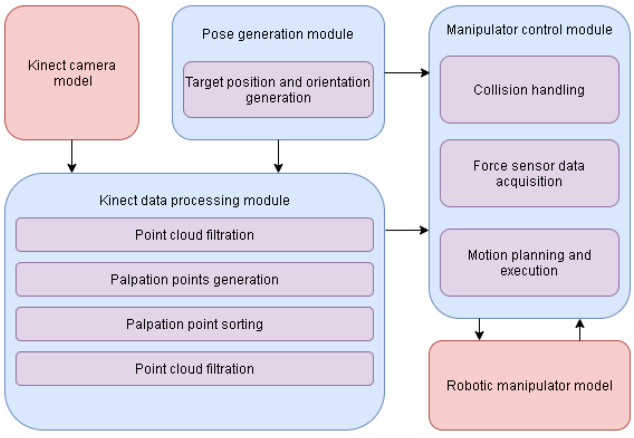

Fig. 4. High level design of the autonomous robotic system model

C. Variable stiffness simulation

While the DART physics engine allows changing stiffness parameters of an entire soft body, it does not support a variable stiffness over the soft body. To overcome this limitation, a simple Gazebo model plugin was developed. A 3D range of high stiffness area coordinates was manually specified. As soon as the manipulator's end-effector entered the high stiffness area, the plugin increased the entire body stiffness; the stiffness was switched back to its original value as soon as the end-effector was leaving the high stiffness area.

III. Autonomous Robotic System

A high-level design of the system is presented in Fig. 4 and consists of a Kinect data processing module, a pose generation module and a manipulator control module. All ROS nodes, which are active during the autonomous palpation simulation, and communications between those nodes are displayed in Fig. 5. The main node "/iiwa/moveit_palpation" operates the entire system and consists of the three aforementioned modules. The following subsections provide more detailed description of each module.

A. Kinect data processing

To process data acquired from the Kinect camera model, Point Cloud Library (PCL) framework [20] was used. Once a point cloud is acquired by subscribing to "/kinect_transform" topic (Fig. 5), the following steps are performed:

1) Coordinate frame transformation to move the point cloud from a Kinect frame to a global frame
2) Invalid points removal
3) The point cloud spatial filtering to remove points that are not in the region of interest
4) The point cloud downsampling to generate points for the palpation
5) The point cloud sorting according to the palpated surface coverage algorithm (for an abdominal palpation a snake scan algorithm was implemented)

Upon completion of the above steps the manipulator control module is able to query a next point for the palpation. As soon as the next point is requested, the Kinect data processing module calculates a normal vector to the palpated surface at a next palpation point location and provides a point position and a normal vector.

B. Pose generation

When a position and an orientation of the manipulator end-effector are required, the pose generation module is used. There are two such cases in the system. The first case is a conversion of a palpated surface point position and a surface normal at that point to the end-effector position and orientation. In this case, the normal vector is used to compute a rotation matrix for the end-effector in order to align it with a vector opposite to the normal vector at a current palpation point. The second case occurs when an indentation step is performed and the indenter should be moved deeper into the palpated body. In this case, the end-effector pose is translated

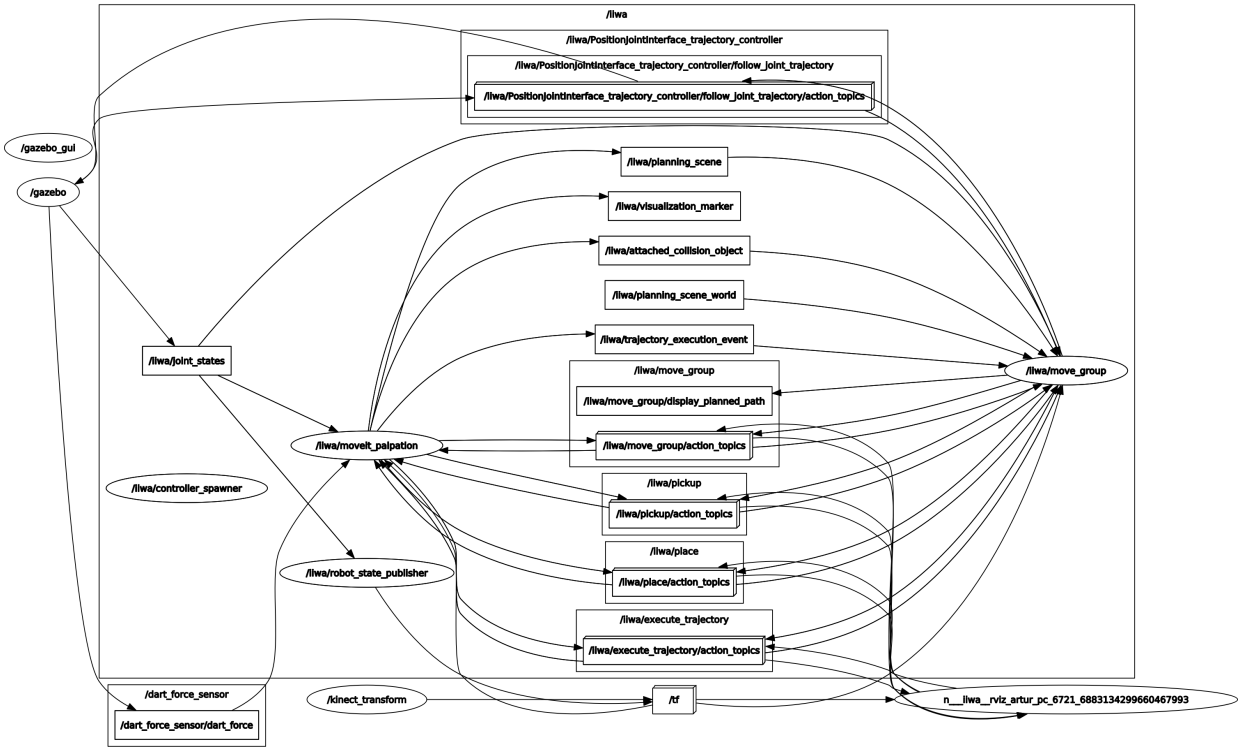

Fig. 5. ROS nodes and communications of the autonomous system

in the direction opposite to the normal vector at a current palpation point keeping a current orientation unchanged.

C. Manipulator control

The manipulator control module uses ROS [21] and MoveIt motion planning framework [22] capabilities for C++ to control the manipulator and handle collisions. Each time the manipulator should move to a new pose, the module performs motion planning and (if the planning succeeds) motion execution. Motion planning, motion execution and current state acquisition functionalities are provided by "/iiwa/move_group" node (Fig. 5) that transfers any required data from or to different MoveIt system components (e.g.,"/iiwa/joint_state" for a current state acquisition).

A palpation should be performed in a systematic manner [23], each time moving to an adjacent area until an entire region of interest is competed. To take into account the aforementioned description, a snake scan with a continuous indentation (Fig. 6) was implemented. As opposed to a raster scan and a point-type indentation, these techniques result in a faster surface coverage, since there is no need to return the end-effector to a scan line beginning and to perform the end-effector indentation and retraction for each palpation point.

A collision scene includes a surgical table model and a human abdomen model. The Kinect camera model and a tripod model are not taken into account since they are outside of the manipulator workspace. Positions and orientations of collision scene objects are acquired by subscribing the manipulator control module to corresponding topics published by the Gazebo simulator; in order to be taken into account by the motion planner these data are published to "/iiwa/planning_scene" topic (Fig. 5). To enable the palpation, collisions between the human abdomen model and the spherical end-effector of the manipulator are allowed.

D. Sensing

Our experiments demonstrated that the default force sensor plugin of the Gazebo simulator is unable to perceive stiffness changes in implemented soft surfaces. Thus, instead of measuring forces applied to a joint (as it was implemented in the default force sensor plugin) we developed a new force sensor plugin. This plugin obtains force data from collisions between the end-effector's link and the palpated body by directly accessing the DART physics engine data. The collected data are published to "/dart_force_sensor/dart_forc" topic (Fig. 5). The manipulator control module subscribes to that topic and processes the data in order to prepare them to a visualization stage.

Normal force data are collected during the surface palpation. Considering the correlation between a point mass stiffness and a force that arises as a result of a point mass deformation (Eq.1), we assume that normal force information is sufficient for differentiation between stiffness variations of the soft surfaces, provided that an end-effector's pose is aligned with a surface normal opposite vector and an indentation depth (that

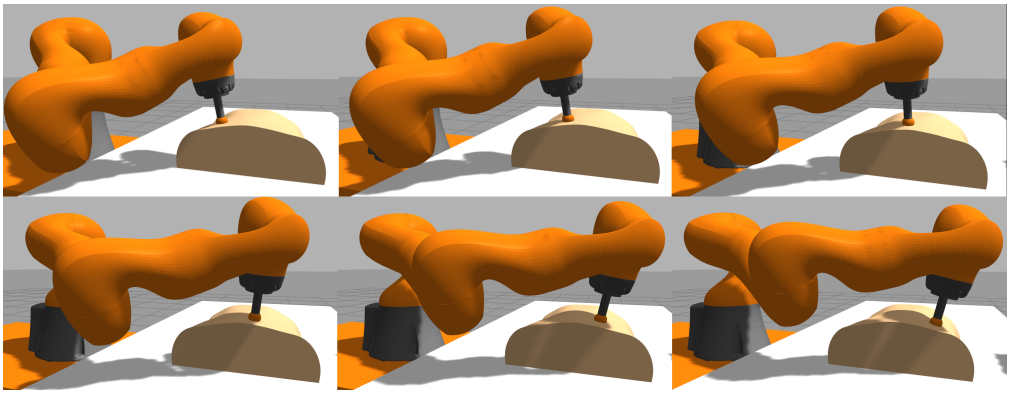

Fig. 6. Different phases of the palpation procedure simulation.

defines deformation in Eq.1) is the same (1 centimeter as in the superficial palpation case) during data gathering phase.

IV. Results

The abdomen, modelled and simulated as a soft object with a variable stiffness over its surface, as described in sections II-B and II-C, was palpated by the autonomous robotic system, presented in section III. By assigning high stiffness to an area, corresponding to a right lower quadrant of the abdomen, a pathological symptom of was simulated. Rigidity of this region might indicate such pathologies as an acute appendicitis, a localized peritonitis, a perforated caecal carcinoma or Crohn's disease [4].

Data, acquired during the palpation, consisted of 3D positions and normal forces, measured at corresponding positions. Due to the original data being highly scattered, a surface fitting operation was performed by raw data interpolation using MATLAB [24] script. Fitted surface provided a visual and a quantitative representation of a normal force distribution over the palpated surface, however the position data became two-dimensional with all palpation points being projected onto a plane.

A resulting normal force map is presented in Fig. 7. The high stiffness area was identified at the correct location (Fig. 8); the region is clearly visible and force values for that region exceed the ones for the rest part of the abdomen model.

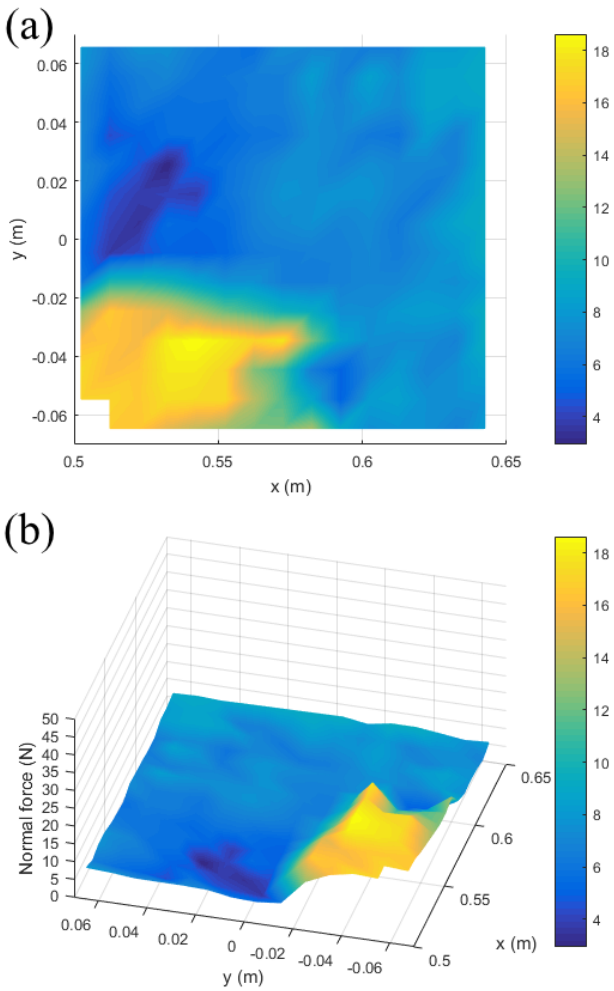

Fig. 7. Normal force distribution over the palpated surface with variable stiffness: (a) two-dimensional view, (b) three-dimensional view.

V. Conclusion

The paper presented the Gazebo model of the autonomous robotic system for a medical palpation. The system is based on KUKA IIWA LBR manipulator model that was modified to enable stable surface contact with a spherical indenter and data collection with our new force sensor plugin. The model utilizes data from the Kinect camera plugin to generate palpation points on a palpated surface. Motion planning and execution were implemented using MoveIt motion planning framework API.

The palpated surface implementation used DART physics engine features with the mass-spring model approach for soft bodies simulation. To deal with DART physics engine limitation in simulating variable stiffness over a soft body, we created a special Gazebo model plugin to simulate the missing functionality. Virtual experiments in the Gazebo demonstrated that the implemented soft bodies provide consistent feedback according to its stiffness parameters and the autonomous

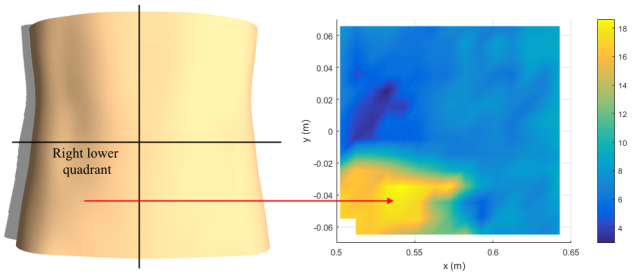

Fig. 8. Correlation of the palpation result (right) and the abdomen soft model with the high stiffness area at the right lower quadrant (left).

system successfully discriminated between areas of different stiffness.

The limitations of the current approach include non optimal implementation of a soft surface, which leads to a low real time factor of the simulation, and relatively simple soft body models in terms of geometry: due to scaling being used as a method of "flesh/bone" structure forming, any excessively protruding part of the model causes "bone" displacements. These issues, together with visualization of deformation feature in the Gazebo, are left as a part of our future work.

Acknowledgment

This work was supported by the Russian Foundation for Basic Research and the Government of the Republic of Tatarstan, project ID 18-48-160037.

References

[1] J. L. Longe, Ed., *The Gale encyclopedia of medicine*, 2nd ed. Detroit, MI: Gale Group, 2002.

[2] C. Jarvis, *Pocket companion for physical examination and health assessment*, 7th ed. St. Louis, MO: Elsevier, 2016

[3] A. Reuben, "Examination of the abdomen," *Clinical Liver Disease*, vol. 7, no. 6, pp. 143-150, June 2016.

[4] A. Cuschieri, P. Grace, A. Darzi, N. Borley and D. Rowley, *Clinical surgery*, 2nd ed. Turin: Blackwell Science Ltd, 2003.

[5] H. Li, W. Liu, K. Wang, K. Kawashima, and E. Magid. "A cable-pulley transmission mechanism for surgical robot with backdrivable capability", *Robotics and Computer-Integrated Manufacturing*, vol. 49, pp. 328-334, 2018.

[6] R. E. Goldman, A. Bajo and N. Simaan, "Algorithms for autonomous exploration and estimation in compliant environments," *Robotica*, vol. 31, no. 1, pp. 71–87, Jan. 2013.

[7] E. Ayvali, A. Ansari, L. Wang, N. Simaan, and H. Choset, "Utility-guided palpation for locating tissue abnormalities," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 864-871, Apr. 2017.

[8] B. Xiao et al., "Depth estimation of hard inclusions in soft tissue by autonomous robotic palpation using deep recurrent neural network," *IEEE Transactions on Automation Science and Engineering* (Early Access), pp. 1-9, Mar. 2020.

[9] J. Guo, B. Xiao and H. Ren, "Compensating uncertainties in force sensing for robotic-assisted palpation," *Applied Sciences*, vol. 9, no. 12, 2573, June 2019.

[10] A. Faragasso et al., "Real-time vision-based stiffness mapping," *Sensors*, vol. 18, no. 5, 1347, May 2018.

[11] H. Li, K. Wang, Z. Li, K. Kawashima and E. Magid. External force estimation of impedance-type driven mechanism for surgical robot with kalman filter. *International Conference on Robotics and Biomimetics (ROBIO)* pp. 2214-2219, 2018. IEEE.

[12] N. Koenig and A. Howard, "Design and use paradigms for Gazebo, an open-source multi-robot simulator," in *IROS 2004*, Sendai, Japan, vol. 3, pp. 2149-2154.

[13] C. Hennersperger et al., "Towards MRI-based autonomous robotic US acquisitions: a first feasibility study," *IEEE Transactions on Medical Imaging*, vol. 36, no. 2, pp. 538-548, Oct. 2016.

[14] M. Pichkalev, R. Lavrenov, R. Safin, K. H. Hsia. Face drawing by KUKA 6 axis robot manipulator. *In 12th International Conference on Developments in eSystems Engineering (DeSE)*, pp. 709-714, 2019. IEEE.

[15] A. Shafikov, A. Sagitov, H. Li, N. Schiefermeier-Mach, and E. Magid, "Robotic palpation modeling for KUKA LBR IIWA using Gazebo simulator," in *ICAROB 2020*, Beppu, Oita, Japan, pp. 436-439.

[16] B. Gabbasov, I. Danilov, I. Afanasyev, E. Magid. Toward a human-like biped robot gait: Biomechanical analysis of human locomotion recorded by Kinect-based Motion Capture system. *In 10th International Symposium on Mechatronics and its Applications (ISMA)*. pp. 1-6, 2015. IEEE.

[17] J. Lee et al., "DART: Dynamic Animation and Robotics Toolkit," *Journal of Open Source Software*, vol. 3, no. 22, 500, Feb. 2018. Accessed on: May 25, 2020. [Online]. Available doi: 10.21105/joss.00500.

[18] S. Jain and C. K. Liu, "Controlling physics-based characters using soft contacts," *ACM Transactions on Graphics*, vol. 30, no. 6, 163, Dec. 2011.

[19] Open Source Robotics Foundation, *SDFormat specification*. Accessed on: May 25, 2020. [Online]. Available: https://sdformat.org/spec

[20] R. B. Rusu and S. Cousins, "3D is here: Point Cloud Library (PCL)," in *ICRA 2011*, Shanghai, China.

[21] M. Quigley et al., "ROS: an open-source Robot Operating System," in *ICRA workshop on open source software*, 2009.

[22] I. A. Sucan and S. Chitta, *Moveit Motion Planning Framework*. Accessed on: May 25, 2020. [Online]. Available: https://moveit.ros.org/

[23] M. E. Z. Estes, *Health assessment & physical examination*, 5th ed. Clifton Park, NY: Thomson Delmar Learning, 2014.

[24] MATLAB, 2016. 9.1.0.441655 (R2016b), Natick, Massachusetts: The MathWorks Inc.