3D Pose Estimation for Laparoscopic Instruments: A Single Camera and a Planar Mirror Approach

Erdal Dayak [1] and Ulus Cevik[2]

[1] Department of Informatics, University of Gaziantep, Gaziantep, Turkey
dayak@gantep.edu.tr
[2] Department of Electrical-Electronics Engineering, Cukurova University, Adana, Turkey
ucevik@cu.edu.tr

Abstract

Minimally Invasive Surgery (MIS) with Laparoscopic instruments (LI) has many advantages over traditional procedures, and thus training of laparoscopic instruments via computer simulations has become important. In this study, a method is proposed for 3d pose estimation. For the proposed scheme, a computer vision based training simulator which used a training box composed of a single camera and a planar mirror was developed. This method has advantages over similar literature methods in terms of processing speed and capability of working even with low resolution images. Furthermore, the proposed approach can handle tool occlusions by using the epiline geometry.

1. Introduction

Minimally Invasive surgery (MIS) or also known as closed surgery has increasingly been used in recent years due to its benefits. It reduces hospital stays and causes less pain when compared to traditional methods [1]. However, this kind of operation is more difficult than open surgeries, since the operation is carried out by only looking at a monitor. Thus, surgeons should receive an additional training to accomplish MIS operations. Therefore, training simulations and learning tools have emerged in order to properly train surgeons for MIS.

In the literature, two major approaches come into prominence: Physical Training Boxes, and Virtual Reality (Simulation) tools. In the former approach, surgeons have the opportunity to improve their hand skills which can be assessed in terms of speed and accuracy using additional software [2], [3]. However, there are some disadvantages in these boxes such as deformation of the training set in time and consumption of some training material as a result of operations like cutting, sewing and separating [4]. As an alternative to physical boxes, Virtual Reality (VR) simulators, which are cheaper and do not require consumption of materials, are also proposed in the literature. When compared to the physical box approach, these methods have some complications in providing the feel of reality and haptic feedback. The VR simulators generally fall into one of the two major categories, sensor-based and computer vision-based simulators. In both techniques, the most significant task is to accurately estimate the 3D pose of the surgery instruments (i.e., LI). To accomplish the estimation, sensor-based simulators utilize magnetic or optic sensors, while computer-vision based ones use image processing techniques. Additionally, sensor-based simulators can be further extended with haptic feedback feature.

In computer vision-based simulators, achieving the 3D pose estimation in real time is a vital task. To this end, some literature studies calculated the pose using more than one camera (e.g., two cameras) to obtain a stereo image, while some others did the estimation by using a single camera view supported with geometric techniques such as vanishing points, perspective projection and so forth. In the study of Allen et al. [5], a method that uses a single camera is proposed. The estimation is based on exploiting vanishing points geometry. According to the reported results, their method fail in some cases such as the case when the instrument comes too close to the camera. In another study, Loukas et al. [6] also proposed a method based on a single camera view. In their study, perspective geometry features are used to estimate the 3D pose. The LI is detected by considering a colored marker put on the distal region of the instrument. This method is reported to have some drawbacks in scenes where occlusion is present, or the marker is partially or completely hidden. Additionally, the error rate of the method increased as the resolution of the image was decreased. Sangkyun et al. [7] utilized a single camera, too. In their paper, three markers were placed on the LI and the 3D pose estimation was done by use of the Haralick algorithm. However, the method did not present a solution to the instrument occlusion problem. Another drawback of the study was that it only worked with a high resolution camera with 1920x1080 at 60 fps. Among similar studies, the paper of Ferdando et al. [8] was one of those that presented a solution to the occlusion problem by using two orthogonal cameras. Their method, however, showed an increased error rate in cases where tracked instrument comes very close to one of the cameras. Moreover, simultaneous use of two cameras resulted in increased processing cost.

Briefly, the literature studies generally have some major problems. Primarily, almost all studies suffer from the occlusion problem. Secondly, when 3D pose estimation is based on a single camera view, some limitations occur. In such studies, either error rate increases or high resolution images are required in order to prevent the error rate from increasing which in return increases the processing load. This study is a hybrid computer vision based approach that uses a single camera. Along with the camera, a planar mirror is used in order to obtain a stereoscopic view which is then processed to accomplish an accurate pose estimation. The proposed approach is a cheaper alternative with respect to similar literature methods using stereoscopic images for estimation. Furthermore, it works faster than those methods using two cameras since it processes only one image while calculating the 3D pose. In the proposed method, a higher accuracy is achieved even in images with lower resolutions. Moreover, processing low resolution images results in low processing times when compared to the methods using two

cameras to obtain stereoscopic views. Therefore, the proposed method requires less computational power when compared to two-camera-based methods.

2. Methods

2.1. The experimental setup

The experiments of the study were conducted in a wooden box in which a camera (Logitech HD C525 WebCam), and a plain mirror were placed (Figure 1). The box was illuminated with LED bulbs. The main body, and the distal region of each LI used in the experiments were marked with different colors. Since the maximum number of LI used in the study was taken to be two, four different marker colors were selected. The height of the main body was 210 mm, while the distal part was 50 mm long. The hardware specifications of the computer on which the experiments were conducted were as follows: Intel® Core™ i7-4702MQ 2.20 GHz (8 CPUs), 1600 MHz DDR3 8GB RAM, NVIDIA GeForce GT 740M video card on Windows 8.1 64-bit operating system.

Figure 1. Controlled experimental setup

2.2. Reflection View and Epipolar Geometry

In the study, a stereo view was obtained by combining the views acquired from the camera, and the planar mirror. Since the image on the mirror was on a coordinate system symmetric to the real view, this image was vertically flipped before further processing.

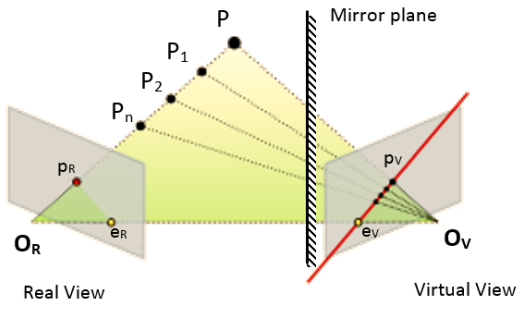

Figure 2. Epipolar geometry via planar mirror reflection

The epipolar geometry was used to determine the relationship between two cameras (one of them is virtual) that observe the same scene from different angles. This geometry is dependent on the relative locations of cameras to each other and the intrinsic parameters of the cameras [9]. While projecting a 3D scene into a 2D image, all points on the 3D space with the same x-y components, and different z values are projected on the same point and thus indistinguishable (Points P_1, P_2... P_n on the O_R-P line in Figure 2 all projected as the point P_r). The other camera that looks at the same scene sees these points with different x-y values (e.g., points on the P_V-e_v line in Figure 2). The line that connects these points in the second camera is called the epiline. Different points in the 3D space produces different epilines. These epilines all cross into a single point called the epipoint (e_v in Figure 2)

2.3. Camera Calibration

In the study, a well-known and widely used camera calibration method, called the Pinhole technique, was adopted. The web camera was calibrated by the use of OpenCV image processing library before the acquired images were processed [10]. Mandatory parameters of camera geometry, such as fx, fy, horizontal/vertical focal length, cy, cx (intrinsic parameters), and the distortion model of the lens were calculated from serial images that contained a chessboard. Therefore, before processing, the pixel coordinates were rectified in accordance with this distortion model. However, the extrinsic parameters ([Rotation | Translation]) should also be determined, as well. These parameters were estimated by the use of a well-known optimization algorithm, named Levenberg-Marquardt [11] that iteratively approximates the required parameters by iterating between points of the circle grid on the camera, and counter-values of these points in the 3D space. This optimization process aims at minimizing the distance between the known reference projection points and the 3D points calculated by the use of 2D points captured from the camera in an iterative manner.

Images of several resolutions (1280x960, 960x720, 800x600, 640x480 and 320x240) were tested in order to observe the effect of resolution over the accuracy of the results. As the default setting, 640x480 resolution was selected through the experiments of the study. A view of the working environment is shown in Figure 1.

2.3. Shaft and tip marker tracking

In the study, HSV (Hue, Saturation, Value) space was preferred in which colors are more easily discriminated by the human eye than they are discriminated in the RGB (Red, Green, Blue) space [12]. The working environment is a controlled environment. The environment, excluding the equipment, is composed of colors white, black, and some tones in between them. For this very reason, a Hue based filter in the HSV space proves to be sufficient in order to filter objects other than the equipment. Under some assumptions, such as the use of a maximum of two equipment, the Hue axis is divided into four regions, each of which is mainly used for a marker (Figure 3-a). Since the HSV filter range in the controlled environment was kept as large as possible, we did not need an adaptive filter model as it was utilized in some literature studies, such as the one by Loukos [6].

2.4. Finding Equipment Center Line

After all images captured from the camera were transformed into the HSV space, a separate filter was used for each marker. Therefore, the shaft and tip of each equipment could be obtained separately. By merging these parts, the equipment can be obtained as a whole (Figure 3-b). The edge lines of the equipment were extracted out of this monochrome image by the use of the Canny edge detection algorithm [13] which proves to

be an effective and well-known algorithm in the relevant literature (Figure 3-c).

The low level information obtained as a result of the edge detection process (pixel level primitive properties) were converted to real lines by the help of Hough transform [14]. Since the Hough transform produces many redundant lines, a second process should take place in order to eliminate these redundancy, and to reveal the sought lines. To accomplish this, all lines obtained from the Hough transform for each equipment (approximately a number of lines between 10 and 30) were analyzed to see if they overlapped with any edges detected by the Canny algorithm. The non-intersecting lines were considered as the correct lines.

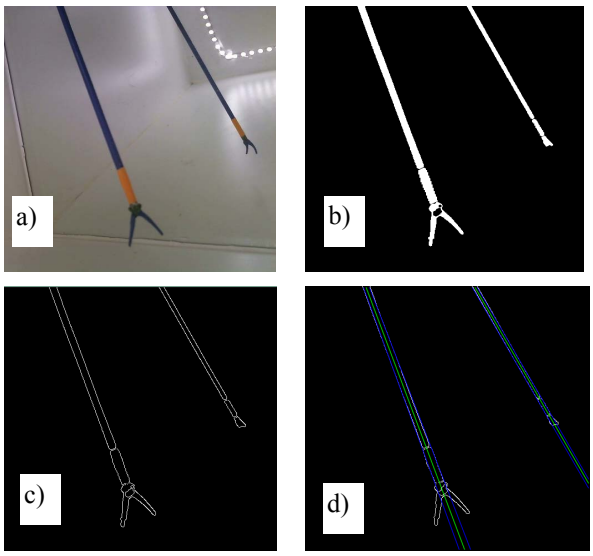

Figure 3. Instrument center line finding progress

a) The original image

b) HSV filter to detect shaft and marked distal of the equipment was applied and two regions were merged

c) Canny Edge detection was applied to the image

d) Hough line detection was applied to find the shaft midline of the equipment

2.5. Finding Tip of Equipment Regions

After the shaft center line was detected, the beginning and ending points of the tip counter were to be calculated. Each contour in the image, which was obtained as a result of the tip marker filter, was calculated by the use of the algorithm proposed by Suzuki et al. [15]. The area of each region was calculated and all regions were ranked according to the size of their area in the reverse order. Then, the minimum rectangle that bounds each region was found by the use of the Toussaint algorithm [16]. The distance from the center of gravity of any of these rectangles to each shaft center lines were calculated, and the closest rectangle to any shaft center line was considered as the tip region of that equipment (Figure 4-b). The upper and lower edges of any rectangle was taken as the beginning and ending points of the tip of the equipment (Figure 4-b). Since these points should be on the trail of the shaft line, the location of these points were corrected by moving them to the closest point to the trail. The beginning and ending points of the tip should be on the line that passes on the epipoint. These lines were checked to see if there was no any deviation, or a reasonable small error occured, by using the view from the real camera, the beginning and ending points of the tip of the virtual image was corrected by aligning it with the epipoint and the shaft line. If this was not the case, and there was an unreasonable amount of error, it was considered that there was an occlusion or an incident of being out of the image. In this case, the points on either the real or the virtual image was taken as the reference, and the other was calculated in accordance with these points (Figure 4-c). The selection between the real, and virtual points was decided by looking at which had the wider angle with respect to the epipoint.

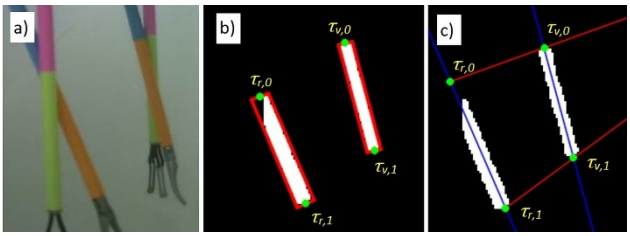

Figure 4. Finding tip begin & end points and correction

2.6. 3D Pose Esitimation

The term 3D pose implies both position and orientation of an object in 3D space. In the case of finding the 3D pose of an equipment's distal region, the 3D coordinates of the beginning and ending points of the tip of the equipment should be found beforehand. Therefore, in the study, the 2D beginning and ending points of the marker (τ_0 and τ_1 in Figure 4), which were obtained by using the stereo view from the real and virtual cameras, were utilized.

Before performing the 3D pose estimation process, these points were undistorted by the use of the fisheye model. Afterwards, these points were multiplied by the inverse of the camera matrix (which held intrinsic parameters) so that the 2D coordinates independent of the camera parameters were obtained. Finally, from these 2D points, the 3D coordinates could be calculated by using the well-known triangulation method given in [17].

2.7. Noise Filter

Through the study, it was observed that the estimated 3D points may shift up to a few pixels between the images captured sequentially, which causes a small vibration effect on the position of the equipment. In order to remove this noise effect and smooth the series of estimated values through sequential images, the Kalman filter [18], a common and proven method in the literature was utilized. This filter can be used on both 2D points and estimated 3D points. Moreover, it is not only suitable for processing pixels values, it can also be applied on any quantitative values that express a state such as points, speed, angle, and so forth. In this paper, the Kalman filter was used to smooth the estimated beginning and ending points of the distal region of the equipment in the 3D coordinate space.

2.8. Virtual Reality Framework

In order to visualize the equipment and establish a virtual scene for user interaction, we utilized a popular framework called SOFA (Simulation Open Framework Architecture) [19].

It is a real-time physical simulation tool which is a very useful particularly for medical simulations. It provides an environment for researchers to develop and test new algorithms in a virtual environment that can simulate interaction of different types of objects under physics rules. Moreover, SOFA has a strong support for rigid and soft body dynamics. In a SOFA virtual scene, each object has three basic components (Figure 5):

- Collision model
- Behavior model
- Visual model

Figure 5. SOFA architecture of an object; left to right: collision, behavior and virtual models

Each component (model) of an object has its sub-types and algorithms. For instance, the collision model could be chosen as the sphere, as well as the point, line and triangular surface. In the study, the models, point, line and triangular surface are all used together. Similarly, the behavior model can be chosen from a range of possible algorithms. In this study, there are two objects in the virtual scene, the equipment and a liver, for which rigid and tetrahedron finite element method behavioral models were used, respectively.

Briefly, SOFA is designed to process the information acquired from the sensors on the equipment through the drivers and to give feedback about the results of the interactions. In order to adapt our application to SOFA environment, a sensable emulator plug-in was developed (e.g. Figure 6).

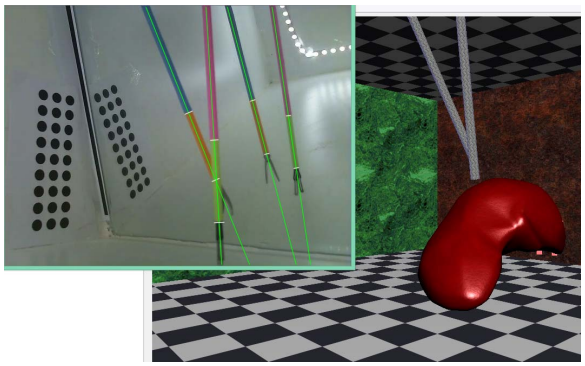

Figure 6. Sofa framework

3. Results and Conclusion

To measure the angular and positional accuracy of the study, the manual test setup, which is shown in Figure 7, was prepared. Accuracy calculations were done in the test setup for over 100 samples each of which was tested at six different resolutions. As a result, an average angular error of 1,02^{o} (± 0,72^{o}) was observed while the maximum error was up to 3,00^{o} (See Figure 8 for details). As for the positional error, it was observed to be 1,04 mm (± 0,83 mm) while the maximum was 4,26 mm (See Figure 9). Notably, the quality of resolution did not affect the angular error while the positional error increased more rapidly as the resolution was decreased (Figure 8 and Figure 9).

The error increased in the particular case that the equipment came very close to the camera. However, considering the average and the maximum error in both angular and positional values, these were not higher than those of literature studies relevant to MIS simulation (e.g. [5], [6]). Also, it was observed that the time to process an image decreased exponentially as the resolution was decreased (Figure 10). The proposed approach was capable of processing at least 30 fps even at the highest resolution of 1280x960. Because the positional error increased rapidly at the resolution of 320x240, the ideal resolution for evaluating the method appeared to be 640x480. At this resolution, the processing speed can be at least 60 fps. All calculations done in the study were accomplished by using a single thread of the CPU and the other cores/threads available in the CPU may be occupied by other tasks. Therefore, the load of the both 3D pose estimation and, MIS simulation calculations can be handled by a single computer.

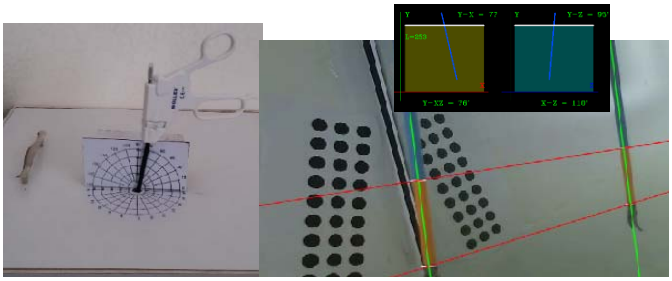

Figure 7. Manuel test setup

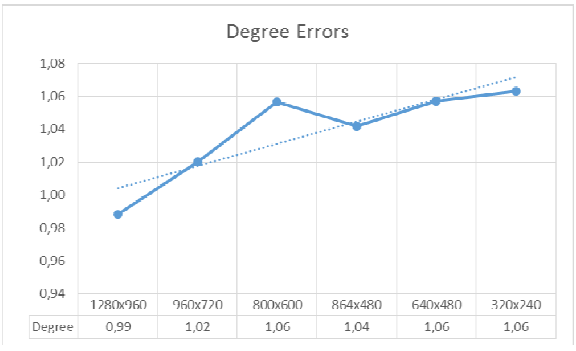

Figure 8. Angular errors for each resolution

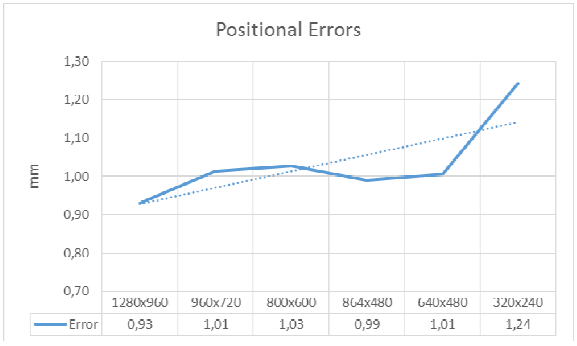

Figure 9. Position errors for each resolution

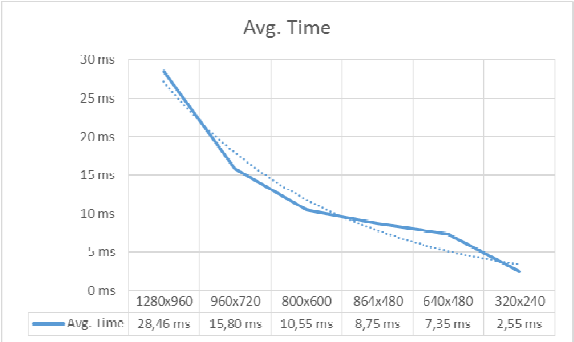

Figure 10. Average calculation time for each resolution.

This study is particularly focused on estimating pitch and yawn angles along with positional information. The roll and grasper openness angles are not considered by the current study, however, future studies may cover these topics, as well. Another restriction of the study is that, up to two equipment are allowed in a scene. Considering the fact that two equipment may not suffice in some MIS scenarios, the proposed approach should be improved in future studies to distinguish more than two equipment by using different HSV ranges or feature filters. One of the additional aspects to be improved is that the interactive scene built by the use of the SOFA framework includes a simple interaction scenario which does not cover some of the common MIS simulation operations. In a future study, we consider to cover more advanced operations, such as surgical actions like cutting-sewing, and more realistic operations on live organ simulations (e.g. operating on a pulsing heart).

4. References

[1] K. H. Fuchs, "Minimally invasive surgery," *Endoscopy*, vol. 34, no. 2, pp. 154–159, 2002.

[2] R. Aggarwal, J. Ward, I. Balasundaram, P. Sains, T. Athanasiou, and A. Darzi, "Proving the Effectiveness of Virtual Reality Simulation for Training in Laparoscopic Surgery:," *Ann. Surg.*, vol. 246, no. 5, pp. 771–779, Nov. 2007.

[3] A. K. Dhariwal, R. Y. Prabhu, A. N. Dalvi, and A. N. Supe, "Effectiveness of box trainers in laparoscopic training," *J. Minimal Access Surg.*, vol. 3, no. 2, pp. 57–63, 2007.

[4] A. E. Park and T. H. Lee, "Evolution of Minimally Invasive Surgery and Its Impact on Surgical Residency Training," *Minim. Invasive Surg. Oncol.*, pp. 11–22, 2011.

[5] B. F. Allen, F. Kasper, G. Nataneli, E. P. Dutson, and P. Faloutsos, "Visual tracking of laparoscopic instruments in standard training environments.," *Stud Health Techol Inf.*, no. 163, pp. 11–17, 2011.

[6] C. Loukas, V. Lahanas, and E. Georgiou, "An integrated approach to endoscopic instrument tracking for augmented reality applications in surgical simulation training," *Int. J. Med. Robot. Comput. Assist. Surg. MRCAS*, vol. 9, no. 4, pp. e34–51, Dec. 2013.

[7] S. Shin, Y. Kim, H. Cho, D. Lee, S. Park, G. J. Kim, and L. Kim, "A single camera tracking system for 3D position, grasper angle, and rolling angle of laparoscopic instruments," *Int. J. Precis. Eng. Manuf.*, vol. 15, no. 10, pp. 2155–2160, Oct. 2014.

[8] F. Pérez, H. Sossa, R. Martínez, D. Lorias, and A. Minor, "Video-based tracking of laparoscopic instruments using an orthogonal webcams system," *World Acad Sci Eng Technol Int J Med Health Pharm Biomed Eng*, vol. 7, no. 8, pp. 184–187, 2013.

[9] G. Xu and Z. Zhang, *Epipolar Geometry in Stereo, Motion and Object Recognition: A Unified Approach*. Springer Science & Business Media, 2013.

[10] G. R. Bradski and A. Kaehler, *Learning OpenCV: [computer vision with the OpenCV library]*, 1. ed., [Nachdr.]. Beijing: O'Reilly, 2011.

[11] J. Moré, "The Levenberg-Marquardt algorithm: Implementation and theory," in *Numerical Analysis*, vol. 630, G. A. Watson, Ed. Springer Berlin Heidelberg, 1978, pp. 105–116.

[12] "HSL and HSV," *Wikipedia, the free encyclopedia*. 21-Apr-2015.

[13] B. Wang and S. Fan, "An Improved CANNY Edge Detection Algorithm," 2009, pp. 497–500.

[14] L. A. F. Fernandes and M. M. Oliveira, "Real-time line detection through an improved Hough transform voting scheme," *Pattern Recognit.*, vol. 41, no. 1, pp. 299–314, Jan. 2008.

[15] S. Suzuki and K. Abe, "Topological structural analysis of digitized binary images by border following," *Comput. Vis. Graph. Image Process.*, vol. 30, no. 1, pp. 32–46, Apr. 1985.

[16] G. T. Toussaint, "Solving geometric problems with the rotating calipers," in *Proc. IEEE Melecon*, 1983, vol. 83, p. A10.

[17] "Triangulation," *Wikipedia, the free encyclopedia*. 22-May-2015.

[18] G. Welch and G. Bishop, "An Introduction to the Kalman Filter," Department of Computer Science University of North Carolina at Chapel Hill, Chapel Hill, TR 95-041, Jul. 2006.

[19] J. Allard, S. Cotin, F. Faure, P.-J. Bensoussan, F. Poyer, C. Duriez, H. Delingette, and L. Grisoni, "Sofa-an open source framework for medical simulation," in *MMVR 15-Medicine Meets Virtual Reality*, 2007, vol. 125, pp. 13–18.