

Computational Tasks in Bronchoscope Navigation During Computer-Assisted Transbronchial Biopsy

Jarosław Bułat¹, Krzysztof Duda¹, Mirosław Socha¹, Paweł Turcza¹,
Tomasz Zielinski², and Mariusz Duplaga³

¹ Department of Measurement and Instrumentation, AGH University of Science and Technology, al. Mickiewicza 30, 30-059 Kraków, Poland
{kwant, kduda, socha, turcza}@agh.edu.pl

² Department of Telecommunications, AGH University of Science and Technology, al. Mickiewicza 30, 30-059 Kraków, Poland
tzielin@agh.edu.pl

³ Collegium Medicum, Jagiellonian University, ul. Skawińska 8, 31-066 Kraków, Poland
mmduplag@cyf-kr.edu.pl

Abstract. The paper presents algorithmic solutions dedicated to computer navigation system which is to assist bronchoscope positioning during transbronchial needle-aspiration biopsy. The navigation exploits principle of on-line registration of real images coming from endoscope camera and virtual ones generated on the base of computed-tomography (CT) data of a patient. When these images are similar an assumption is made that the bronchoscope and virtual camera have approximately the same position and view direction. In the paper the following computational aspects are described: correction of camera lens distortion, fast approximate estimation of endoscope ego-motion, reconstruction of bronchial tree from CT data by means of their segmentation and its centerline calculation, virtual views generation, registration of real and virtual images via maximization of their mutual information and, finally, efficient parallel and network implementation of the navigation system which is under development.

1 Introduction

Virtual bronchoscopy [1] CT-guided approach represents a modern solution to the difficult problem of bronchoscope positioning during medical procedure of transbronchial needle-aspiration biopsy. It makes use of real-time registration of real 2D images (coming from an endoscope) and virtual ones (obtained from virtual camera looking inside a 3D model of bronchial tree, reconstructed from CT patient data by means of segmentation). Usually, the registration of these two-source images is performed using in-the-loop maximization of their: correlation [2] or mutual information [3]. In order to speed-up search for precise virtual camera position, coarse estimation of bronchoscope camera can be performed.

It is usually done from video stream using corresponding points and epipolar geometry [2] or optical flow methods and perspective geometry [4]. In turn, next position of the endoscope camera can be predicted and tracked with Kalman using [5] or Monte Carlo [6] particle filters. Using shape-from-shading technique or triangulation by means of corresponding points it is also possible to extract 3D model of the airways tract from the real-time endoscopic video and try to register it to the 3D model reconstructed from the CT scans. First such attempt has been reported in [7] and further elaborated in [8].

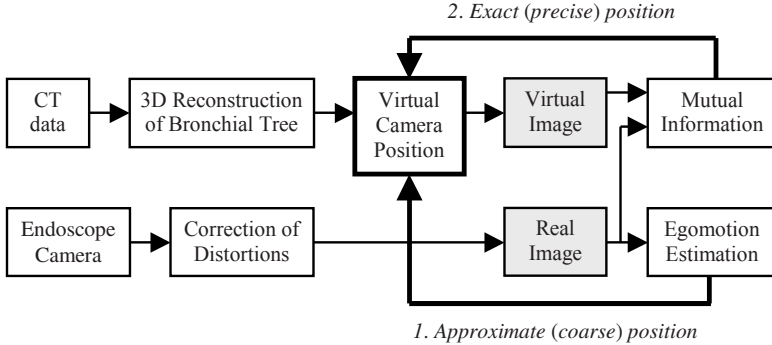


Fig. 1. Block diagram of the system under development

In the paper, some computational and implementation issues concerning the described above bronchoscope navigation scheme are presented. The following aspects are addressed: correction of camera non-linearities, fast approximate estimation of endoscope ego-motion [9], reconstruction of bronchial tree from CT data by means of their segmentation and its centerline calculation [10], virtual views generation, registration of real and virtual images via maximization of their mutual information [11] and, finally, efficient parallel and network implementation of the whole navigation system. The architecture of the system being designed is presented in Fig. 1.

2 Correction of Camera Distortions

The methodology presented in [12], [13] was used for correction of bronchoscope Olympus BF-160 camera lens distortions. As a test image black dots lying on straight lines have been chosen. The distorted image obtained from bronchoscope camera is presented in Fig. 2a. The applied correction algorithm was based on maximization of the criterion measuring the degree of lines straightness. Using the model of radial distortions, the following polynomial was found:

$$r_c(r) = 4.2009 * 10^{-8} r^4 + 1.5991 * 10^{-10} r^3 + 3.7892 * 10^{-13} r^2 + r \quad (1)$$

relating the radius r_c in distorted image to the radius r in undistorted image. For better results, unlike [12], [13], in our approach the center of distortions was

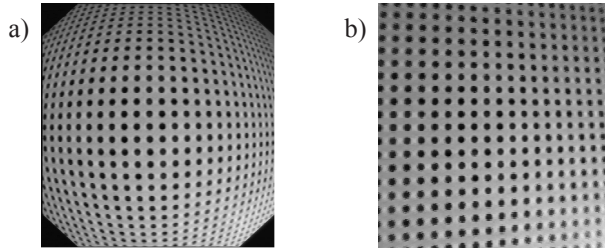


Fig. 2. a) Test image from bronchoscope camera, b) Reconstructed, undistorted image

calculated by means of set of optimizations in the neighborhood of geometric center of the image. The image after correction is presented in Fig. 2b.

3 Segmentation of Bronchial Tree and Calculation of Navigation Path

Image segmentation is the operation of grouping image pixels into separate objects present in a picture. The first step of segmentation algorithm is most often feature extraction and then checking if the specific pixel belongs to the object of interest. In medical CT data segmentation is used to isolate biological structures of interest like whole organs, e.g. bronchial tree, or some interesting, smaller structures like lymphatic nodes or tumors. Segmentation algorithms can be divided into four major groups: *pixel-based*, *region-based*, *edge-based* and *model-based* methods. In our research, the airway tree was segmented with the following steps: data smoothing with 3D gaussian filter, global thresholding and checking 26-connectivity. The exemplary segmentation results are depicted Fig. 3.

Centerline of the segmented bronchial tree is used as a navigation path in: virtual bronchoscopy, planning transbronchial biopsy and guiding bronchoscope's tip during biopsy. Therefore its precise computation is very important. Classification of algorithms for automatic generation of centerline (navigation path) in bronchial tree can be found in [14], [15].

We have proposed a new algorithm based on the distance transform, acting on the segmented bronchial tree, and an original iterative method for path searching [10]. The procedure is equipped with additional heuristic rules that prevent detecting false paths. The algorithm for path detection starts with placing the cube at the beginning of the bronchial tree with sides parallel to CT data coordinates. Fig. 3 shows the position of the cube in bronchial tree during successive steps and values of distance transform on its sides. The transform values on the cube sides are used for setting up the next point of the path. In the case depicted in Fig. 3a, the distance transform shows that the next point of the path should be either in Z or $-Z$ direction. As the direction $-Z$ means going back to previously computed point, the direction Z is chosen. In case depicted in Fig. 3b, from possible $-Z$, Y and $-Y$ the direction Y is used, while the direction $-Y$ is stored and becomes

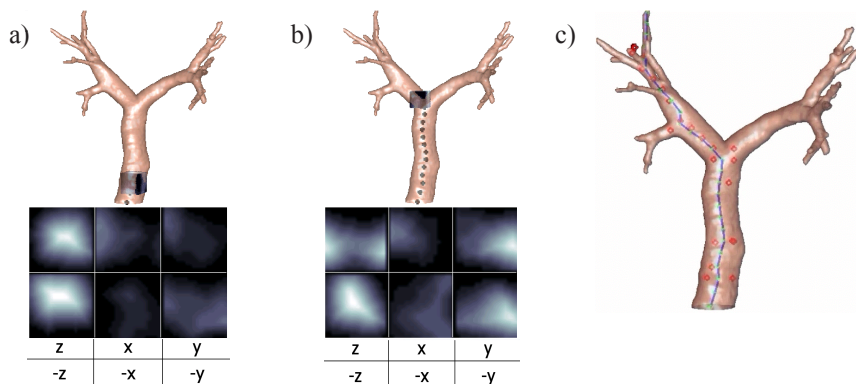


Fig. 3. a), b) Successive steps of computing centerline (navigation path) in bronchial tree (at the bottom values of computed distance transform lying on cube sides are shown); c) Path computed after the first iteration (points connected by the line) are used as starting points for next iterations (points not connected).

starting point for the next iteration (possible branching node) and the direction $-Z$ is neglected for the same reason as previously. Fig. 3c presents result of the first iteration: calculated points of the navigation path that are connected with lines. Consecutive iterations start at points stored as possible branching nodes. The algorithm ends up after checking all branching nodes (what takes 49 iterations in the presented example). Finally, a polynomial of 6-th degree was fitted to calculated points in order to make trajectory smoother.

4 Visualization of Bronchial Tree – Virtual Phantom

The visualization part of the navigation system was developed with Borland C++ Builder and Visualization ToolKit (VTK) cross-platform, open-source library [16]. The Visualization ToolKit makes use of the OpenGL API for 3D graphic card. The surface rendering technique was used for the sake of good performance and quality of generated virtual bronchoscopy images. Surface rendering includes two stages: generation of three-dimensional surface representing bronchial tree walls from CT data and visualization process via a graphic card. Virtual bronchoscopy images (VB) were used for testing motion estimation and navigation algorithms based on image registration.

The process of 3D surface generation consists of: loading a DICOM file with patient's computed tomography (CT) data, cropping CT data to reduce their size and generating isosurface at the level of -500HU by means of marching cubes algorithm [17]. The isosurface on this level goes through the data that represent walls of patient's bronchial tree. The result of computations is the continuous triangular mesh. The triangle strips are created in order to improve rendering performance. For these data a mapper was created to generate OpenGL rendering

primitives and actor object for controlling mesh property which is the final step for virtual bronchoscopy image generation. The generated surface can be saved to file for later use.

In order to achieve maximum resemblance with real bronchoscope camera illumination conditions, the virtual light source was set up as follows: it moves along with the camera and its position is the same as camera position. The light is configured as positional (headlight), and the light cone angle corresponds to camera cone angle. To prevent overexposing of nearest surfaces the irregular light intensity along the cone angle was assumed. Light fading attenuation was also used for distance simulation.

5 Fast Estimation of Bronchoscope Egomotion

In order to speed-up egomotion estimation [18] in bronchial environment we use simplified model of geometric relations based on cylindrical shape accompanied by the fixation on a carina [19], what reduces motion's degrees of freedom to four (forward/backward movement, camera rotation, camera tilt in two directions). It is achieved by continuous tracking of the carina (stationary point) illuminated by the camera light source, and by analyzing bronchial wall radial moves relative to fixed point by correlation in polar coordinates [9].

Reverse perspective projection of images before correlations is made by the use of correspondence between z -axis and r -axis derived from the following trigonometric relation (see Fig. 4):

$$\frac{R - r}{z} = \operatorname{tg} \varphi = \frac{R}{z - f} \quad \implies \quad r = R \left(1 - \frac{z}{z - f} \right) \quad (2)$$

Let us note that R serves only as a scaling factor of the view. In the current research we estimate forward motion, after carina stabilization and camera rotation compensation, as arithmetic mean of directional wall motions. Camera tilt is estimated from geometric mean of these motions.

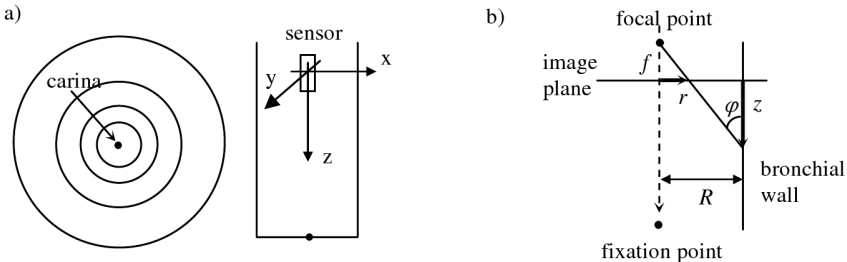


Fig. 4. Applied models: a) segment of bronchial tree (upper perspective projection and x - z cross-section), b) imaging in cylindrical environment with radius R , camera focal length f , radial image axis r and depth from image plane z .

We assessed algorithm accuracy by series of test in virtual cylinders, virtual bronchial trees and on real operational video sequences from transbronchial biopsy [11]. The results of experiments show that accuracy of bronchoscope cumulated motion estimation is within 5% of distance in virtual environments. In Fig. 5a the virtual bronchial tree environment with estimated wall motions is shown. Fig. 5b shows estimation of cumulated forward/backward motion together with imposed forward virtual camera motion. Disturbing factors in this experiment were camera rotation, and x-y plane camera moves.

Fig. 6a shows forward/backward bronchoscope trajectory during real biopsy. This trajectory suggest similarity of frames 7 and 65, being distant in time but close in space, because of the strong backward move followed by the forward move. These frames, shown in Fig. 6b and 6c, confirm this similarity and confirm also satisfactory behavior of our egomotion estimation algorithm.

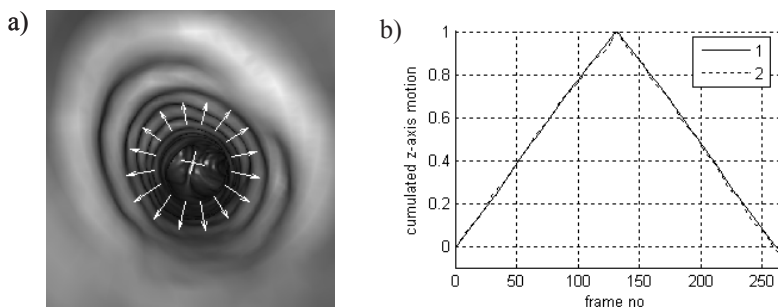


Fig. 5. Example of camera position estimation along z-axis in virtual bronchial-tree phantom: a) Virtual environment with estimated radial wall-move vectors, b) Estimated forward/backward camera trajectories for imposed motion: 1 - camera motion along the path with target fixed on carina, 2 - camera motion with additional camera tilt and rotation and moving target.

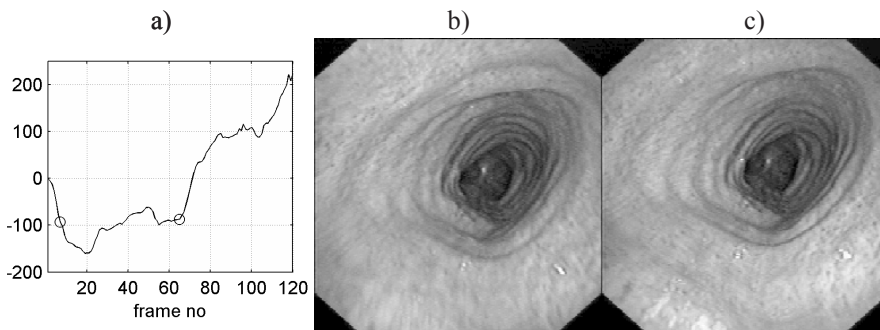


Fig. 6. Estimation of camera position along z-axis from real data: a) estimation result with two frames close in space but distant in time (mark 'o'), b), c) frames 7 and 65

6 Image Registration Using Mutual Information

The information from egomotion algorithm of bronchoscope motions is not sufficient to precise determination of the location of *real* bronchoscope tip in relation to *virtual* bronchial tree, however, can significantly speed-up navigation process. Therefore, before successful navigation will be possible, two tasks have to be completed. The first one is to place the *virtual bronchoscope* (the source of virtual images) in a position corresponding to the real bronchoscope. This is achieved by adjusting position of virtual bronchoscope in such a way that generated images are similar as much as possible to images from real bronchoscope.

After setting up the virtual camera starting position, the second task - calibration of egomotion estimation algorithm is performed. Having two images from real camera at positions z_0 and $z_0 + d$, where z_0 is the starting position and d is outcome of egomotion estimation algorithm, using appropriate image similarity measure we try to find such a displacement of the virtual camera position which makes virtual image as similar to the real one as possible. Egomotion estimation is used for coarse estimation of virtual camera position, then image registration algorithm is used for finer adjustment.

Methods enabling registration of images from the same or different sources have been extensively developed through the last decades. Numerous papers were published on this topic [20]. In our approach, in both above described tasks, mutual information [21] was used as an image similarity measure. It is based on the concept of joint entropy as given by Shannon for determination of communication's channel capacity and is defined as follows:

$$I(u, v) = H(u) - H(u | v) \quad (3)$$

where $H(u)$ denotes the measure of uncertainty of the value of random variable u , and $H(u | v)$ denotes the same measure but determined with the assumption that value of random variable v is known. In this way $I(u, v)$ expresses how much the uncertainty about value of u decreases after getting to know value v . Correlation between decreasing value of conditional entropy $H(u | v)$ and increasing value of mutual information $I(u | v)$ is obvious. Using the Bayesian theorem: $P(A, B) = P(A | B)P(B)$ and the definition of Shannon's entropy

$$H(u) = - \sum_i p_u(i) \log p_u(i), \quad H(u, v) = - \sum_{i,j} p_{uv}(i, j) \log p_{uv}(i, j), \quad (4)$$

the equation expressing mutual information (MI) may be rewritten into the form

$$I(u, v) = H(u) + H(v) - H(u, v). \quad (5)$$

It includes joint entropy $H(u, v)$, which may be determined on the basis of joint probability distribution, which in turn can be inferred from the joint histogram $h(u, v)$ after appropriate normalization.

Exemplary images from real and corresponding virtual camera are presented in Fig. 7. This figure also shows values of mutual information as a function of virtual camera position. In the experiment, the virtual camera was shifted along the

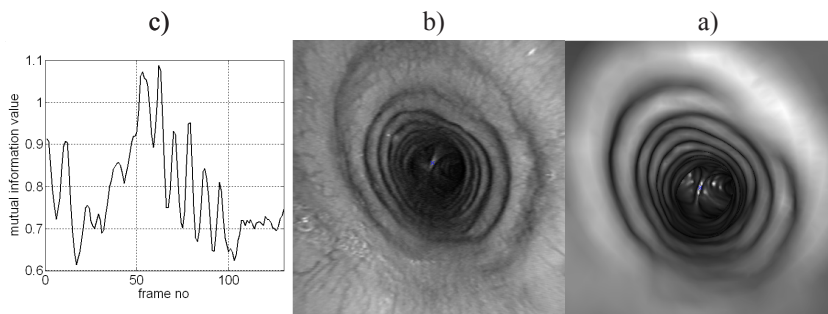


Fig. 7. Example of image registration: a) Mutual information as a function of virtual camera position, b) Real image - frame 71, c) Virtual image corresponding to the real one found by the registration algorithm

computed navigation path lying in the central part of the airways. Observed local maxima of the mutual information curve comes from bronchial tree vertebrae.

7 Computation Complexity Analysis

System presented in Fig. 1. have been implemented in Matlab language with the exception of 3D image generation realized by means of hardware supported OpenGL. In spite of using fast matrix calculation and JIT (just-in-time) optimization in the latest version of Matlab, its real-time operation is not possible on a high-end x86 class computer.

Since the video frame rate from bronchoscope is 25 frames per second, we have only 40ms for accomplish one cycle of the proposed navigation algorithm. Execution time of the most important navigation system procedures programmed both in Matlab and C/C++ language is shown in Table 1.

For precise bronchoscope motion estimation it is necessary to perform: one camera correction, one coarse bronchoscope motion estimation and on average 25 virtual image generations and mutual information calculations. For that reason one cycle of complete motion estimation needs approximately 5600 ms in Matlab and 620 ms in C/C++. One can see that even for optimized C/C++ version it is not possible to perform the algorithm in real-time.

Table 1. Estimated execution time of the most important navigation system procedures for Pentium 4 3.2 GHz processor (single thread)

Procedure	Matlab time [ms]	C/C++ time [ms]
Camera correction	13	1.4
Coarse motion estimation - egomotion	4600	460
3D image generation (one image)	–	1.4
Mutual information (one image)	46	4.9

There are, however, many possibilities for further execution time optimization. First of all, we can use SSE (Streaming SIMD Extensions) instructions of x86 CPUs instead of plain C/C++. BLAS (Basic Linear Algebra Subprograms) [22] seems to be the most efficient way of using SSE extension. Next, we can take advantage of parallel nature of the presented above navigation algorithm using relatively cheap multicore and multiprocessor systems. Both most demanding procedures (egomotion estimation and mutual information calculation) are highly independent within one cycle of motion estimation. What is more, BLAS library has been recently ported to GPU environment and could take advantage of up to 128 independent processors cores that was introduced in the G80+ generation of the NVIDIA GPUs family with CUDA architecture.

In order to take advantage of massive parallel computation we have chosen MPI (Message Passing Interface) as a inter-process communication framework. MPICH2 (implementation of MPI v2 protocol) provide us a great performance and excellent flexibility. It is available on many hardware and software platforms including Linux and Windows as well as provide network transparency. Using this open source library we are able to split visualization process and computation engine to different computer connected by TCP/IP network. During testing procedure of data (image) and control commands transmitting over MPI we have achieved the following results: up to 1GB/s of messages with data frame between two processes on one computer (four core Xeon), up to 500000 short (128 bytes) control messages per second in the similar setup and up to 95Mbps between two processes on two different computer over 100Mbps Ethernet.

Obtained results have convinced us that it is possible to build described above system working in real-time on the basis of standard PC architecture.

8 Conclusions

In the paper some computational and implementation issues and new solutions for bronchoscope navigation during computer-assisted transbronchial biopsy were presented. In authors opinion the proposed, new, very simple and fast algorithm for real-time estimation of endoscope forward and backward ego-motion is the most crucial in its precise positioning using virtual bronchoscopy CT-based approach. After successful simulation tests real-time implementation of the described navigation system is under development at present. We have already estimated partial algorithms' complexity of this modules and tested some particular hardware solutions: multiprocessor NVIDIA CUDA architecture and MPI-based communication framework which provide high performance parallel environment with minimum development effort.

References

1. Bartz, D.: Virtual Endoscopy in Research and Clinical Practise. Eurographics - Computer Graphics Forum 24(1) (2005)
2. Mori, K., Deguchi, D., Sugiyama, J., et al.: Tracking of a Bronchoscope Using Epipolar Geometry Analysis and Intensity-Based Image Registration of Real and Virtual Endoscopic Images. Medical Image Analysis 6(3) (2002)

3. Sherbondy, A.J., Kiraly, A.P., et al.: Virtual Bronchoscopic Approach for Combining 3D CT and Endoscopic Video. In: Medical Imaging 2000, Proc. SPIE (2000)
4. Helferty, J.P., Higgins, W.E.: Combined Endoscopic Video Tracking and Virtual 3D CT Registration for Surgical Guidance. In: Proc. IEEE ICIP, pp. II-961–964 (2002)
5. Nagao, J., Mori, K., et al.: Fast and Accurate Bronchoscope Tracking Using Image Registration and Motion Prediction. In: Barillot, C., Haynor, D.R., Hellier, P. (eds.) MICCAI 2004. LNCS, vol. 3217, pp. 551–558. Springer, Heidelberg (2004)
6. Deligianni, F., Chung, A., Yang, G.-Z.: Predictive Camera Tracking for Bronchoscope Simulation with CONDensation. In: Duncan, J.S., Gerig, G. (eds.) MICCAI 2005. LNCS, vol. 3749, pp. 910–916. Springer, Heidelberg (2005)
7. Bricault, I., Ferretti, G., Cinquin, P.: Registration of Real and CT-Derived Virtual Bronchoscopic Images to Assist Transbronchial Biopsy. IEEE Trans. on Medical Imaging 17(5), 703–714 (1998)
8. Deligianni, F., Chung, A., Yang, G.-Z.: pq-Space Based 2d/3D Registration for Endoscope Tracking. In: MICCAI 2003. LNCS, vol. 2878, pp. 311–318. Springer, Heidelberg (2003)
9. Twardowski, T., Zieliński, T., Duda, K., Socha, M., Duplaga, M.: Fast estimation of broncho-fiberscope egomotion for CT-guided transbronchial biopsy. In: IEEE Int. Conference on Image Processing ICIP-2006, Atlanta (2006)
10. Duda, K., Duplaga, M.: Automatic generation of a navigation path for virtual bronchoscopy. In: PAK, vol. 5bis, pp. 115–118 (2006) (in polish)
11. Duda, K., Zieliski, T., Socha, M., et al.: Navigation in bronchial tree based on motion estimation and mutual information. In: ICSES, Poland (2006)
12. Vijayan Asari, K., Kumar, S., Radhakrishnan, D.: A New Approach for Nonlinear Distortion Correction in Endoscopic Images Based on Least Squares Estimation. IEEE Trans. on Medical Imaging 18(4), 345–354 (1999)
13. Helferty, J.P., Zhang, C., McLennan, G., Higgins, W.E.: Videoendoscopic Distortion Correction and Its Application to Virtual Guidance of Endoscopy. IEEE Trans. on Medical Imaging 20(7), 605–617 (2001)
14. Kiraly, A.P., Helferty, et al.: Three-Dimensional Path Planning for Virtual Bronchoscopy. IEEE Trans. on Medical Imaging 23(9), 1365–1379 (2004)
15. Bartz, D., Mayer, D., et al.: Hybrid segmentation and exploration of the human lungs. In: IEEE-Visualization-2003 (No. 03CH37496), pp. 177–184 (2003)
16. VTK – The Visualizer Toolkit, <http://www.vtk.org/>
17. Lorensen, W.E., Cline, H.E.: Marching cubes: a high resolution 3D surface construction algorithm. Comput. Graph. 21, 163–169 (1987)
18. Tian, T.Y., Tomasi, C., Heeger, D.J.: Comparison of Approaches to Egomotion Computation. In: IEEE Conf. CVPR, pp. 315–320 (1996)
19. Daniilidis, K.: Fixation simplifies 3D Motion Estimation. Computer Vision and Image Understanding 68(2), 158–169 (1997)
20. Zitova, B., Flusser, J.: Image registration methods: a survey. Image and Vision Computing 21, 977–1000 (2003)
21. Pluim, J.P.W., Maintz, J.B.A., Viergever, M.A.: Mutual information based registration of medical images: a survey. IEEE Trans. on Medical Imaging 22, 986–1004 (2003)
22. BLAS (Basic Linear Algebra Subprograms), software library, <http://www.netlib.org/blas/>