U.P.B. Sci. Bull., Series C, Vol. 75, Iss. 3, 2013 ISSN 2286 – 3540

CT IMAGE PROCESSING IN HIP ARTHROPLASTY

Anca MORAR[1], Florica MOLDOVEANU[2], Alin MOLDOVEANU[3], Victor ASAVEI[4], Alexandru EGNER[5]

The use of dedicated software for medical image processing and analysis has become a common practice. One of the domains where medical image processing can be of real help to doctors is hip arthroplasty. Using X-rays and CT images, doctors have to choose the most appropriate implant to insert in the femur at the level of the hip. The prosthesis replaces some dysfunctional parts of the bones. Computers can be very helpful in the analysis of the medical images, before and after hip surgeries. They can also be used in the process of prototyping the implants based on CT images, highly increasing the quality of the medical procedure, because in this case the implant is personalized. The problem with the automatic processing of CT images is their poor quality regarding the low contrast between bones and other tissues and the inhomogeneity within bones. This paper proposes a new method for the segmentation of bones in CT images and the reconstruction of the femoral bone. These processing steps represent parts of a software system whose aim is to produce 3D models of personalized prostheses. The proposed method can highly reduce the time spent for the preparation of the surgeries in hip arthroplasty.

Keywords: CT bone segmentation, bone reconstruction, hip arthroplasty

1. Introduction

Nowadays computers have proven to be useful tools in many domains and medicine does not make an exception. Medical image processing can be used for diagnosis and assistance in treatment prescribing, for the visualization of possible results after surgeries (for example in plastic surgeries), for assistance during surgeries, but also for following the patient’s evolution after surgeries. In this context we propose a new method for the processing of CT images in order to help doctors in the hip arthroplasty field. One of the biggest problems with the

---

[1] Assist., Dept. of Computer Science, University POLITEHNICA of Bucharest, Romania, e-mail: anca.morar@cs.pub.ro

[2] Prof., Dept. of Computer Science, University POLITEHNICA of Bucharest, Romania, e-mail: florica.moldoveanu@cs.pub.ro

[3] Assoc. Prof., Dept. of Computer Science, University POLITEHNICA of Bucharest, Romania, e-mail: alin.moldoveanu@cs.pub.ro

[4] Lect., Dept. of Computer Science, University POLITEHNICA of Bucharest, Romania, e-mail: victor.asavei@cs.pub.ro

[5] Assist., Dept. of Computer Science, University POLITEHNICA of Bucharest, Romania, e-mail: alexandru.egner@cs.pub.ro

majority of medical images is their poor quality, due to the low contrast between image components representing different human tissues (for example, muscles and bones) and the noise introduced by the image acquisition devices. There are further imperfections, like those caused by the position shifting of a patient during the scan, but these will not be taken into account because they rarely occur.

The human eye can easily distinguish features in medical images even in case of very poor quality (blurred and noisy images). For example, a radiologist can differentiate very well between bones and other tissues in a CT image. But how long does it take a human specialist to segment the bones from a whole CT dataset (for example a dataset with 256 slices)? Computers can help with the speedup of the process. Medical software can interpret images much faster than a human being. They can also provide a way of analyzing the whole scanned body (a 3D image), not only a section through the body (a 2D image). A CT dataset is a stack of 2D images, but with the help of specialized software the 3D model of the scanned body parts can be obtained. The 3D model can be further processed in a way that would be difficult or even impossible without the use of a computer. For example, after obtaining the 3D model of a femur, the axis of the femoral body can be determined, or the volume occupied by the femoral head can be computed.

The ultimate goal of the research described in this paper is to develop a software system capable to automatically generate the 3D model of a personalized prosthesis, starting from a stack of CT femoral images. The second chapter of this paper provides a general presentation of hip arthroplasty and the use of CT images. The third chapter discusses the past work that was conducted by our team in this field and presents several other image processing techniques and software applications that can be used in arthroplasty. The forth chapter proposes a new algorithm for 3D segmentation, which extracts the volumes occupied by bones (in this particular case, femoral bones) in CT image. The fifth chapter shows some results regarding the computing times and the obtained 3D models. The sixth chapter presents the conclusions and the future possible research directions in order to automatically or semi-automatically generate the 3D model of a personalized artificial implant that can be produced by a prototyping device.

2. Medical background

This chapter provides some general concepts of hip arthroplasty and shortly describes the use of medical images in this particular field.

2.1. Arthroplasty – general presentation

Hip arthroplasty [1] is a surgical procedure in which the arthritic or dysfunctional joint at hip level is replaced with prosthesis or by remodeling or realigning the joint. Fig. 1 presents a femur and a pelvic bone, before and after inserting an artificial implant.

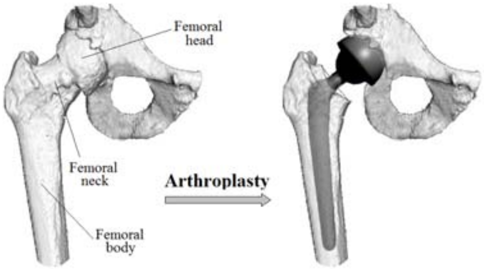


Fig. 1. Hip arthroplasty: before (left) and after (right) inserting a prosthesis

The parts of the femoral bone that are of interest in hip arthroplasty are the head, the neck and the body (as in Fig. 1). Based on the characteristics of these bone parts (dimension, position, etc), a prosthesis is inserted in the interior of the femoral body. The neck and the head of the femur are replaced by the neck and the head of the prosthesis.

2.2. Medical images in hip arthroplasty

Doctors make decisions regarding the insertion of prostheses based on medical images. X-rays (radiographic images) are used in the pre-operatory stage in order to decide whether a patient needs surgery, based on some measurements at hip level [2,3]. But X-rays cannot provide a very good description of the 3D characteristics of the bones since they are only a projection of the scanned body. A CT dataset consists of a stack of 2D images that represent sections through the scanned body. Fig. 2 shows the connection between X-rays and CT images.

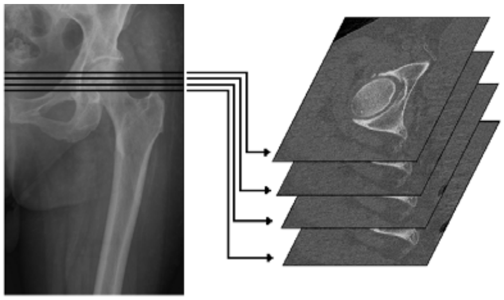


Fig. 2. A vertical view of the human body at the level of the hip provided by a radiographic image (left) and a stack of CT images (right) that represent sections through the scanned body

3. Previous work in medical image processing and hip arthroplasty

Among the research projects in medical image processing some of the most relevant are:

- SCANIP [9], an image processing software that provides a broad range of image analysis and visualization tools for medical purposes.
- 3D-DOCOTOR [10], a project that comes with an advanced 3D modeling software, with strong processing and measurement functions for medical images.
- Hip-OpCT [11], a software that allows importing CT images in DICOM format. Once imported, the CT dataset is visualized through several modalities from which the doctors can plan the surgeries in hip arthroplasty.

We have worked on a project whose aim was to assist doctors in hip arthroplasty. The first contribution in this field was a software for femoral and hip radiographic image processing. This software makes, automatically or semi-automatically, some measurements which otherwise have to be done manually by doctors on radiographic images. [2] and [3] propose some algorithms based on Canny edge detector and Hough transform that automatically extract important parameters in hip replacement from X-ray images.

[4] describes a semi-automatic method for the 3D simulation of fitting prostheses inside femoral bones. But the user interaction is time consuming. This paper proposes a fully automated pipeline for the final goal of obtaining personalized implants.

The first problem is the segmentation of bones from other tissues. The poor quality of the CT images poses some difficulties to the existing segmentation algorithms. [5] presents a new segmentation algorithm based on active contours without edges. This algorithm takes into account all of the characteristics of CT images in order to differentiate between bones and other tissues, but also between different bones.

The next subsection shortly describes the new segmentation algorithm as compared to other techniques.

3.1. New segmentation algorithm based on active contours without edges

The algorithm proposed in [5] has seven steps that are briefly described:

Step 1. Gaussian filter and active contours without edges: this step is based on active contours without edges proposed by Chan and Vese [6]. It uses an evolving curve, the frontier between foreground and background, in a manner that minimizes the differences in intensity within the foreground and within the background. Instead of segmenting the image into a binary image like in [6], this step only enhances the contrast between the foreground and the background.

Step 2. Gaussian and nonlinear anisotropic diffusion: this step is applied to smooth the resulting image, but with the preservation of the edges.

Step 3. Adaptive threshold: this step uses two thresholds (T_1 and T_2) that separate the pixels into three categories: background pixels (with the intensity lower than T_1), foreground pixels (with the intensity greater than T_2) and candidate pixels. The pixels in the third category are filtered with an adaptive threshold in order to be labeled as object or background pixels.

Step 4. Adaptive threshold and Gaussian: the fourth step applies another adaptive threshold, but this time on the initial image.

Step 5. Combining images: the output images from the third and forth steps are combined, obtaining an image with very few incorrect labeled pixels.

Step 6. Island extraction: in this step the foreground pixels that are connected to each other are grouped into islands. The islands are further referred to as slice islands, because an island belongs to one slice. Based on the average intensity of the pixels contained in the current slice island, it is decided whether that island is removed from the foreground or not.

Step 7. Hole filling step: the last step solves the problem of inhomogeneity within the foreground. It starts from the first background pixel in the image and does a breadth first search (BFS) in order to extract all the background pixels connected with that one. All the other background pixels that were not visited in the BFS search are re-labeled as belonging to the foreground.

Fig. 3 presents the output of each step of the segmentation algorithm.

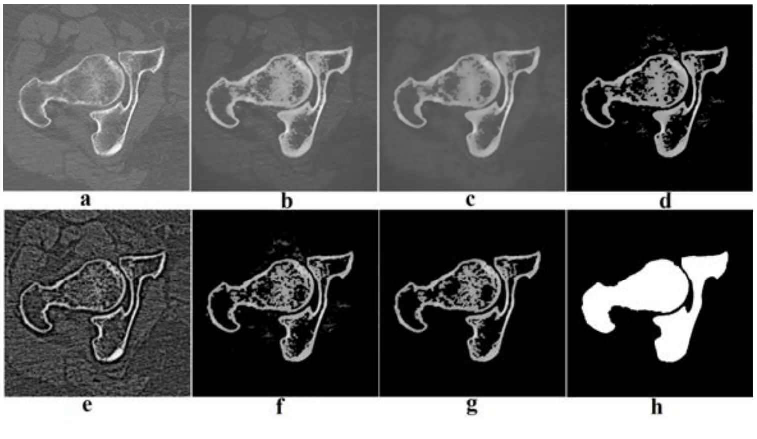


Fig. 3. Output images in the steps of the segmentation algorithm: initial image (a), output of Step 1 (b), output of Step 2 (c) , output of Step 3 (d), output of Step 4 (e), output of Step 5 (f), output of Step 6 (g), output of Step 7 (h)

3.2. Comparison with other segmentation algorithms

This new segmentation algorithm was compared in [5] with three other segmentation techniques, active contours without edges [6] and two implementations based on graph cuts [7]. The tests were conducted on ten noisy CT datasets. The error of incorrectly labeled pixels ("bones" or "other tissues") and the percentage of slices where two objects were wrongly connected were taken into account. Our approach was better than the three segmentation algorithms (but comparable with active contours) regarding the labeling of pixels. But, unlike the other methods, our method discriminated quite well between different bones.

4. Bone reconstruction algorithm

This chapter proposes a new method of bone reconstruction from CT images which is composed of two stages: the segmentation of the bones in CT images and the reconstruction from the segmented images (in this particular case, the reconstruction of the femur). The input of the bone reconstruction stage is composed of the binary segmented CT images into foreground (bone) and background (other tissues) pixels. Even if any bone segmentation algorithm can be incorporated in this method, we use the technique presented in the previous chapter, slightly altered. Fig. 4 presents the steps of the bone reconstruction algorithm.

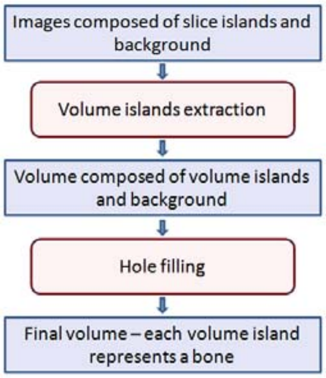


Fig. 4. Bone reconstruction process

4.1. Alteration to the segmentation phase

The last step in the segmentation phase, i.e., the hole filling, has a disadvantage that was discussed in [5]. The method starts a BFS search from the first encountered background pixel and visits all the background pixels that are connected to the first one. All the other background pixels that are not visited in the BFS are considered as being part of the object. This step re-labels all the background pixels that are surrounded by slice islands, either representing inhomogeneities within the foreground or real cavities inside the object. A possible solution for differentiating between inhomogeneities and cavities is given in [5]. If there is the certainty that some pixels belongs to real cavities, and not to inhomogeneities within the object, these pixels can be set as other seed points for the BFS that searches for all the connected background pixels. These seed points could be found based on the intensity. However, in CT images, the difference in intensity between inhomogeneities and cavities is sometimes insignificant. Fig. 5(a) provides such an example. As can be observed, the bone marrow has a similar intensity with the tissue that is located between the femoral bone and the pelvis. Another solution would be a user interaction scenario where the doctors would choose the seed points belonging to the cavities. But the user interaction is time consuming. This chapter proposes an automatic solution for this problem that moves from the processing of each slice individually to a technique that takes into account the relationship between slices. The algorithm proposed in [5] is altered by the removal of the hole filling step. Thus, the input for the bone reconstruction phase is the segmented dataset, composed of the slice islands and the background. An example of a slice from an input dataset is given in Fig. 3(g) and Fig. 5(b).

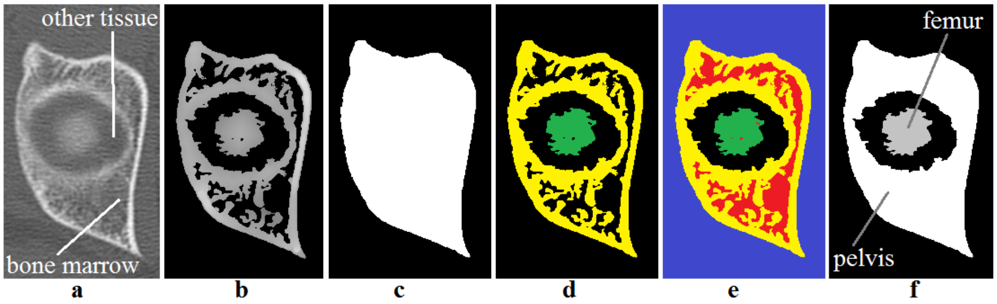


Fig. 5. Hole filling step: (a) initial image, (b) result of island extraction (Step 6) from segmentation algorithm, (c) result of hole filling (Step 7) from segmentation algorithm, (d) result of volume island extraction: there are two volume islands, one marked with green and one marked with yellow, (e) illustration of hole filling step in the bone reconstruction phase, (f) correct segmentation obtained by applying the hole filling step in the bone reconstruction phase

4.2. Volume islands extraction

This step consists of connecting slice islands from adjacent slices, belonging to the same objects, forming volume islands. A volume island consists of all the slice islands from the dataset that are considered to be neighboring islands. The concept of neighboring islands is further described.

Let SI_1 and SI_2 be two slice islands from adjacent images in the CT dataset and let $size(SI)$ be the number of pixels belonging to a slice island SI. Two foreground pixels from adjacent slices $m_1(x_1,y_1,z_1)$ and $m_2(x_2,y_2,z_2)$ are neighbors if and only if:

$$\begin{cases} x_1 = x_2 \\ y_1 = y_2 \\ |z_1 - z_2| = 1 \end{cases} . \tag{1}$$

Let $connection(SI_1,SI_2)$ be the number of neighboring pixels pairs (m_1,m_2) where $m_1 \in SI_1$ and $m_2 \in SI_2$. A rule for deciding whether two slice islands are neighbors was experimentally determined. If

$$connection\ (SI_1, SI_2) > \min(size(SI_1), size(SI_2)) \cdot \beta \ , \tag{2}$$

where $0 < \beta < 1$ is a fixed parameter, then SI_1 and SI_2 are considered neighboring slice islands. In the tests we have chosen the value of $\beta = 0.25$. This value is big enough to ensure that islands from different objects do not wrongly connect, and small enough to ensure that islands belonging to the same object are considered neighbors.

The next step in the workflow is to delete the volume islands with the size (number of pixels) smaller than a given threshold T. T is chosen so that only the volume islands composed of noise pixels are deleted. In our tests the variable T was set to 10000. This does not erase bone tissue, since the bone volume islands are composed of tens of thousands of pixels.

4.3. Hole filling

After identifying all the volume islands, the hole filling step can be applied on each slice individually, as illustrated in Fig. 5(e). The first stage consists of a BFS starting from a background pixel . It can be assumed, without loss of generality, that the seed point of the BFS is a background pixel that does not belong to the interior of a slice island. The pixels discovered with this BFS are marked with blue in Fig. 5(e). All the background pixels that were not visited in the BFS are further inspected. The connected background pixels that were not visited form background islands. A background island is considered to be connected with a slice island if it has at least one pixel that is a direct neighbor of a pixel from that slice island. If a background island is connected with only one slice island then its pixels are re-labeled as foreground pixels. This condition ensures that pixels belonging to other tissues besides bone do not connect different volume islands representing bones. The pixels marked with red in Fig. 5(e) form islands that are connected to a single volume island, either the green or the yellow one. They are re-labeled as being part of the foreground. The pixels marked with black are connected to both the green and the yellow island. They remain labeled as part of the background.

The extraction of a background island and its inspection requires a BFS, starting from the first encountered pixel that belongs to the island.

5. Results

The bone reconstruction algorithm was tested on six noisy CT datasets representing body scans at the level of the hip. Each dataset contained slices where pixels belonging to different bones (the femur and the pelvis) were close to each other. Also, four datasets had a series of images where an object (part of the femur) was located inside another object (part of the pelvic bone), as in Fig. 5. Fig. 6 shows a slice from each CT dataset.

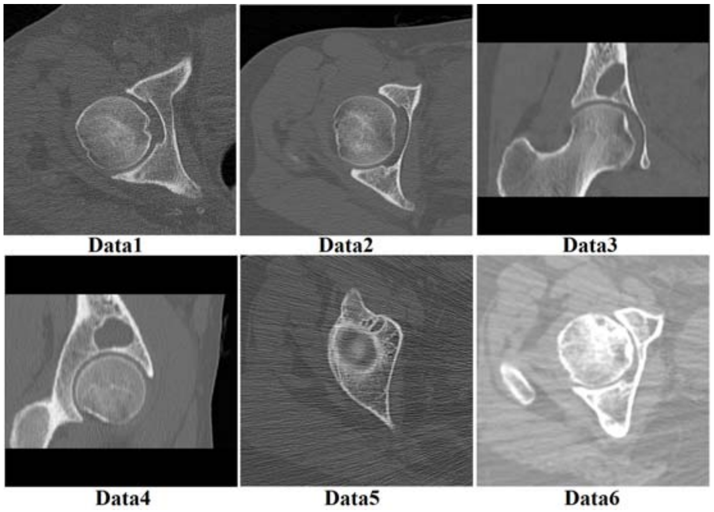

Data1 Data2 Data3
Data4 Data5 Data6

Fig. 6. Slices from the tested datasets

Due to the noise in the last two datasets Data5 and Data6, some user interaction was required. After the segmentation phase, a percentage of slices (approximately 8%) still contained pixels wrongly connecting different objects. These slices were manually modified, by removing the wrongly labeled foreground pixels. Table 1 shows the time required for the user interaction process. Table 1 also presents the computing times for the volume islands extraction and the hole filling step. The processing in the bone reconstruction stage is quite fast. The hole filling takes longer than the volume islands extraction because it performs several breadth first searches for each slice.

Table 1

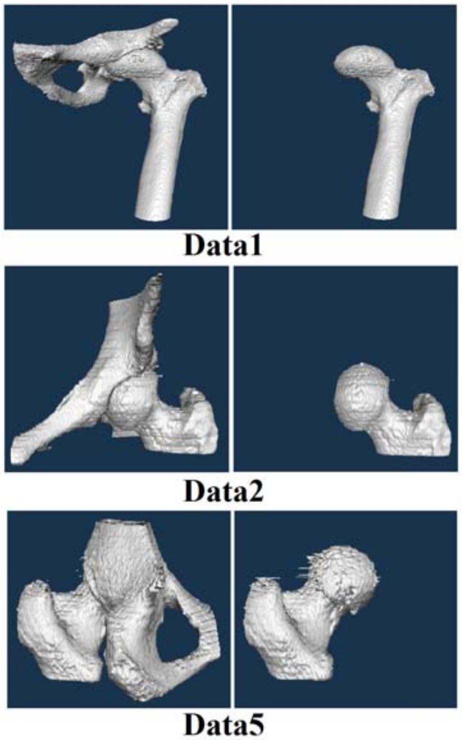
User interaction time for altering the output of the segmentation phase and computing times for the bone reconstruction process

Dataset Size	Data1 $258x512^2$	Data2 $88x512^2$	Data3 $10x512^2$	Data4 $10x512^2$	Data5 $86x512^2$	Data6 $86x512^2$
User interaction (sec)	-	-	-	-	300	360
Volume islands extraction (sec)	1.61	0.57	0.08	0.08	0.58	0.57
Hole filling (sec)	82.99	28.93	3.08	3.02	27.94	28.16

For the last two datasets that were very noisy, the most time consuming operation was the user interaction of adjusting the output generated in the bone segmentation phase. However, it must be mentioned that for the first four datasets there was no user interaction needed. The chosen segmentation method was the best as compared to the original active contours without edges algorithm and two segmentation algorithms based on graph cuts, regarding the time consumed in the user interaction process. A detailed comparison between the proposed segmentation technique and the other mentioned algorithms is given in [5]. If there is previous knowledge regarding the number of bones scanned in a CT dataset and the method identifies the same number of bones, there is no user interaction required. On the contrary, if the number of bones automatically identified by the segmentation and reconstruction method is smaller than the actual number of bones, it means that the software segmentation wrongly connected different bones. In this case, the user has to correct the segmentation error.

After obtaining masks for all the volume islands (each volume island represents a different bone), the isosurface describing the 3D model of the bones can be reconstructed using the marching cubes algorithm [8]. Fig. 7 presents the output of surface reconstruction from three of the datasets. After identifying different masks for the two bones, the user can decide which bone will be rendered. All three datasets contain parts of the femoral bone and the pelvic bone. In the left-side images, both masks are used, resulting in the rendering of both the femur and the pelvis. In the right-side images, only the mask of the femur is used.

Fig. 8 presents the output of surface reconstruction performed on one of the datasets. In the left image, both masks are used. In the right image, only the mask of the pelvic bone is used.

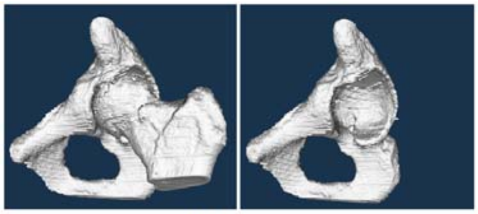


Data1

Data2

Data5

Fig. 7. Surface reconstruction with marching cubes from CT datasets, after applying the bone reconstruction algorithm: in the left images both the femur and the pelvis are rendered; in the right images only the femur is rendered



Data6

Fig. 8. Surface reconstruction with marching cubes from a CT dataset, after applying the bone reconstruction algorithm: in the left image both the femur and the pelvis are rendered; in the right image only the pelvic bone is rendered

6. Conclusions

This paper describes a new method that can be used in the process of automatically or semi-automatically generating personalized implants in hip arthroplasty. The first stage of the method is the segmentation of the bones from CT datasets. For this task we chose the segmentation technique described in [5]. The segmentation method was slightly altered in order to provide better output for the next stage of our method, the bone reconstruction. The bone extraction process solved the problems of the hole filling step from the segmentation stage, by taking into account the relation between adjacent slices. The next step in the generation of personalized prostheses could be the extraction of the interior of the femur (the volume occupied by bone marrow). After that, the skeletonization of the volume occupied by the femoral bone marrow could be performed. The skeletonization would help in computing a series of parameters [2,3]. For example, after obtaining the skeleton of the femoral bone, the position of the femoral body axis, the femoral neck axis and the diameter of the femoral body can be determined. These measurements can be useful in the generation of customized artificial implants.

REFERENCES

[1]. *Robert E. Kennon,* Hip and Knee Surgery: a Patient’s Guide to Hip Replacement, Hip Resurfacing, Knee Replacement and Knee Arthroscopy, Middlebury, CT: Orthopaedic Surgery, P.C., 2008.

[2]. *A. Morar, F. Moldoveanu, A. Moldoveanu, V. Asavei, A. Egner.* “Computer Assisted Analysis of Orhopaedic Radiographic Images”, in Proceedings of the 9th WSEAS International Conference on Signal Processing, Catania, 2010.

[3]. *A. Morar, F. Moldoveanu, A. Moldoveanu, V. Asavei, A. Egner,* “Medical Image Processing in Hip Arthroplasty”, in WSEAS Transactions on Signal Processing, **vol. 6**, issue 4, 2010.

[4]. *A. Morar, F. Moldoveanu, A. Moldoveanu, V. Asavei, A. Egner,* “Computer Assisted Insertion of Prostheses Based on Medical Images”, in 18th International Conference on Control Systems and Computer Science, Bucharest, 2011.

[5]. *A. Morar, F. Moldoveanu, E. Gröller,* “Image Segmentation Based on Active Contours without Edges” (unpublished), accepted at 8th International Conference on Intelligent Computer Communication and Processing, Cluj, 2012.

[6]. *T. F. Chan, L. A. Vese,* “Active Contours without Edges”, in IEEE Transactions on Image Processing, issue 2, pp. 266-277, 2001.

[7]. *Y. Boykov, O. Veksler,* “Graph Cuts in Vision and Graphics: Theories and Applications”, in Math. Models of C. Vision: The Handbook, Springer, 2006.

[8]. *W. E. Lorensen, H. E. Cline,* “Marching Cubes: a High Resolution 3D Surface Construction Algorithm”, in SIGGRAPH ’87 Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques, USA, 1987.

[9]. SCANIP software: Available at http://www.simpleware.com/software/scanip/ last visited in 04.07.2012

[10] 3D-DOCTOR project: Available at http://www.ablesw.com/3d-doctor/ last visited in 04.07.2012

[11] Hip-OpCT software: Available at http://www.hipop.it/hipopct.html last visited in 04.07.2012