MATHEMATICAL BIOSCIENCES
AND ENGINEERING
Volume **xx**, Number **0xx**, xx **20xx**

doi:10.3934/mbe.2009.xx.xx

pp. **1–xx**

FAST TWO DIMENSIONAL TO THREE DIMENSIONAL REGISTRATION OF FLUOROSCOPY AND CT-SCANS USING OCTREES ON SEGMENTATION MAPS

Luca Bertelli

ECE Department, University of California, Santa Barbara
Santa Barbara, CA 93106, USA

Frédéric Gibou

Computer Science and Mechanical Engineering Departments, University of California, Santa Barbara
Santa Barbara, CA 93106, USA

(Communicated by Erik Bollt)

Abstract. We introduce a computationally efficient approach to the generation of Digital Reconstructed Radiographs (DRRs) needed to perform three dimensional to two dimensional medical image registration and apply this algorithm to virtual surgery. The DRR generation process is the bottleneck of any three dimensional to two dimensional registration system, since its computational complexity scales with the number of voxels in the Computed Tomography Data, which can be of the order of tens to hundreds of millions. Our approach originates from the segmentation of the volumetric data into multiple regions, which allows a compact representation via Octree Data Structures. This, in turn, yields efficient storage and access of the attenuation indexes of the volumetric cells, required in the projection procedure that generates the DRR. A functional based on Mutual Information is then maximized to obtain the alignment of the DRR with the two dimensional X-ray fluoroscopy scans acquired during the operation. Promising experimental results on real data are presented.

1. **Introduction.** In the context of minimally invasive surgical procedures, such as those in the lungs, surgeons employ the tracking of surgical instruments in conjunction with preoperative or intraoperative images in order to indirectly guide the procedure. Often, a detailed three-dimensional description of the body structures can easily be constructed *before* surgery using high resolution CT-scans. *During* the actual surgical procedure, other modes of imaging are used, such as two dimensional X-ray fluoroscopy, because they are fast, cheap and less toxic. The goal of image guided surgery is to locate and track the position of the instruments during surgery thanks to two dimensional imaging techniques and then project the result back onto the three dimensional model. One of the main computational challenges is the three dimensional to two dimensional registration algorithm to recover the Pose (defined as the position, orientation and scale of three dimensional volumes) and the subsequent tracking algorithms designed to follow the surgical instruments.

In order to compare comparable entities, the registration of a three dimensional volume to a two dimensional image requires an intermediate step: The construction of a so-called DRR, i.e. the two dimensional image obtained by casting virtual

2000 *Mathematics Subject Classification.* Primary: 00A72, 68U10; Secondary: 65N50.
Key words and phrases. Digital Reconstructed Radiography, Octree Mesh Generation.

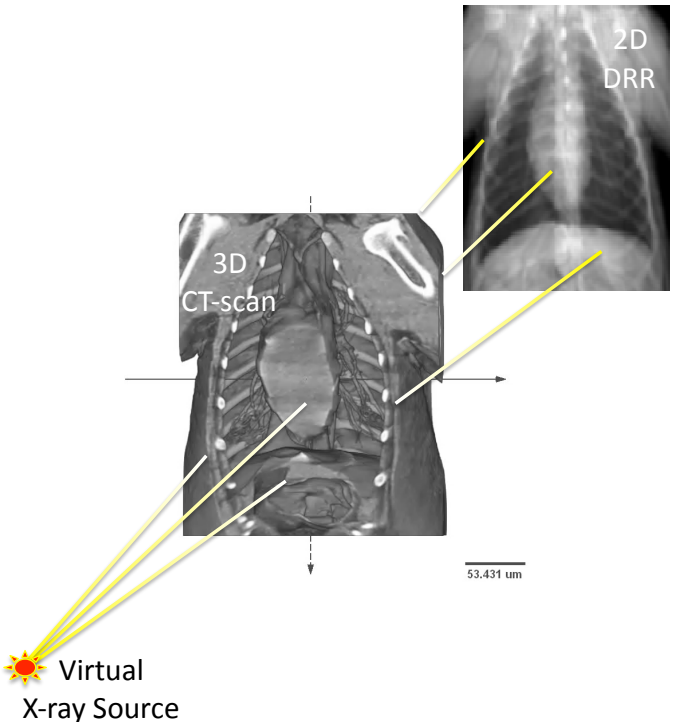


FIGURE 1. Schematic illustrating the virtual ray casting procedure used to generate a Digitally Reconstructed Radiograph. The 2D DRR in this figure is that obtained with the algorithm described in the present paper.

X-rays through the three dimensional scan and measuring their attenuation due to the attenuation index of the voxel they intersect (see figure 1). This process is meant to mimic the physical process of obtaining two dimensional X-ray scans by casting X-rays through a patient's body. Once the DRR is constructed for a given orientation of the three dimensional volume, it is possible to measure the difference in some norm between the DRR and the two dimensional X-ray fluoroscopy scan. This process can be repeated with different orientation of the three dimensional volume and an optimization algorithm can be applied to register (or align) the DDR with the target X-ray fluoroscopy scan.

The generation of DRRs is computationally expensive since the complexity scales with the total number of voxels in the three dimensional CT data. This can therefore be the bottleneck of the entire registration system. Since real-time procedures are sought, fast algorithms need to be designed. Several algorithms have been proposed in the recent past (see e.g. [17, 8] and the references therein) to address this problem, but this is still believed to be an open problem. The difficulty stems from the fact that both efficiency and accuracy must be considered. In all cases, most of the effort is focused on reducing the computational complexity of the algorithm in order to be able to obtain a real time computation of the registration parameters. In such case, the three dimensional data can be maintained registered onto the X-ray image during surgery, allowing for possible motion of the patient. In [17], the authors used

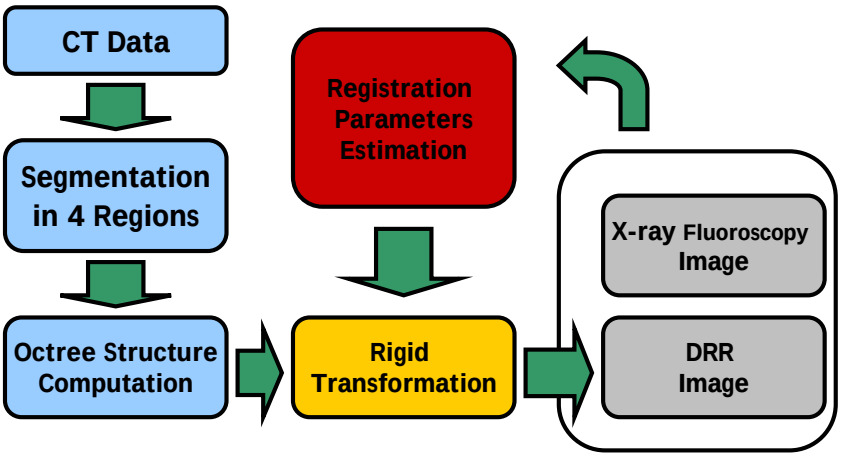


FIGURE 2. Different steps of the three dimensional to two dimensional registration procedure.

a sparse sampling of the DRR image for the computation of the image statistics, in such a way that the complete generation of the DRR is not required. In [8], the authors used an extension of light field rendering technique from the computer graphics community [4] to generate the DRR. We take a different approach where we exploit the optimal representation of the Octree data structure combined with the segmentation of the CT-data into four regions.

2. **Outline of the Algorithm.** We propose an approach based on segmentation and we demonstrate how the CT data, which consists mostly of high quality images, can be segmented in an effective way, which allows then to describe the three dimensional structure in a very efficient way via Octree representation. This procedure does not only makes extremely more compact the description, but also allows a very fast access to the content of the three dimensional structure. This, in turn, becomes crucial in speeding up the ray casting procedure required to create the DRR. We detail the algorithm in this paper and provide preliminary results when it is applied to real data. The proposed method can be summarized as follows, where the first three steps focus on the construction of the DRR, the main focus of this paper, and where the last step is an example of registration:

1. **Segmentation of the CT data into 4 regions**: bones, fat, muscle and lungs/air. The step is performed using a modified version, extended to 4 regions, of the method presented in [3], along with the regularization procedure of [1].
2. **Representation of the three dimensional CT data** using an Octree data structure, where each leaf node (three dimensional cell) contains the average gray level of the CT-scan in that cell. The segmentation result is used to identify the regions in space where high resolution is needed to accurately represent the regions' boundary and the regions where a coarse representation is sufficient. This step can be divided into two separate sub-steps:

- Three dimensional Fast Marching Method (FMM) to compute the distance of each three dimensional voxel to the closest boundary given by the segmentation algorithm.
- Construction of the Octree given the distance map output of the FMM.

3. **Construction of the Digital Reconstructed Radiograph**. This step consists in casting virtual rays through the CT data to simulate the X-ray fluoroscopy procedure in order to obtain a two dimensional DRR image that can be registered onto the X-ray fluoroscopy image. The Octree structure computed in the previous step reduces significantly the number of cells visited and therefore the computational time.
4. **Registration of the Fluoroscopy Image to the DRR**. In this phase the pose parameters are estimated to maximize a functional based on Mutual Information, as in [17].

The different steps are graphically represented in figure 2, where the complete registration procedure is described using bounding boxes. In the following sections we describe each step and we present experimental results obtained on real data.

3. **Level Sets Based Segmentation .** Three-dimensional image segmentation can be cast as a surface evolution problem in the sense that surfaces, initialized at random on the data, are evolved until an optimal configuration is reached. This optimal configuration is usually defined as the minimum of a particular cost function. Following [7], a surface Σ, the boundary of an open set $R_i(\Sigma) \in V$ (*i.e.* $\Sigma = \partial R_i(C)$), is implicitly represented as the zero level set of a continuous Lipschitz function $\phi : V \mapsto \mathbb{R}$. The function ϕ is positive for the points within the set $R_i(C)$ and negative elsewhere (*i.e.* for the points within $R_o(C) = V \backslash R_i(C)$). Therefore the Heaviside function $H(\phi)$, along with its complementary $\big(1 - H(\phi)\big)$, can serve as indicator functions for the points in $R_i(C)$ and $R_o(C)$ respectively [2]:

$$\chi_1 = H(\phi) \quad = \quad \begin{cases} 1 & \text{if } \phi > 0 \\ 0 & \text{elsewhere} \end{cases} \tag{1}$$

$$\chi_2 = \big(1 - H(\phi)\big) \quad = \quad \begin{cases} 1 & \text{if } \phi < 0 \\ 0 & \text{elsewhere} \end{cases} \tag{2}$$

In [15] the authors described how k level set functions can be used to construct up to $n = 2^k$ different indicator functions and therefore to represent up to n different regions. Using this convention, in the case of $n = 4$ regions, we can write the four characteristic functions as:

$$\chi_1 = H(\phi_1)H(\phi_2) = \begin{cases} 1 & \text{if } \phi_1 > 0 \text{ and } \phi_2 > 0 \\ 0 & \text{elsewhere} \end{cases}$$

$$\chi_2 = H(\phi_1)\Big(1 - H(\phi_2)\Big) = \begin{cases} 1 & \text{if } \phi_1 > 0 \text{ and } \phi_2 < 0 \\ 0 & \text{elsewhere} \end{cases}$$

$$\chi_3 = \Big(1 - H(\phi_1)\Big)H(\phi_2) = \begin{cases} 1 & \text{if } \phi_1 < 0 \text{ and } \phi_2 > 0 \\ 0 & \text{elsewhere} \end{cases}$$

$$\chi_4 = \Big(1 - H(\phi_1)\Big)\Big(1 - H(\phi_2)\Big) = \begin{cases} 1 & \text{if } \phi_1 < 0 \text{ and } \phi_2 < 0 \\ 0 & \text{elsewhere} \end{cases}$$

Using this notation, [2, 15] designed a level-set segmentation algorithm based on the Mumford-Shah functional. In particular, the energy functional:

$$E(\phi_1, \phi_2) = \sum_{i=1}^{4} \int_V f_i(\mathbf{x}) \chi_i(\mathbf{x}) d\mathbf{x} \tag{3}$$

is minimized. In this energy, $f_i(\mathbf{x}) : V \mapsto \mathbb{R}$ is an indicator function of how well pixel $\mathbf{x}$ fits in the region denoted by χ_i. The functions f_i are defined exploiting prior information about the four regions to be segmented. Here

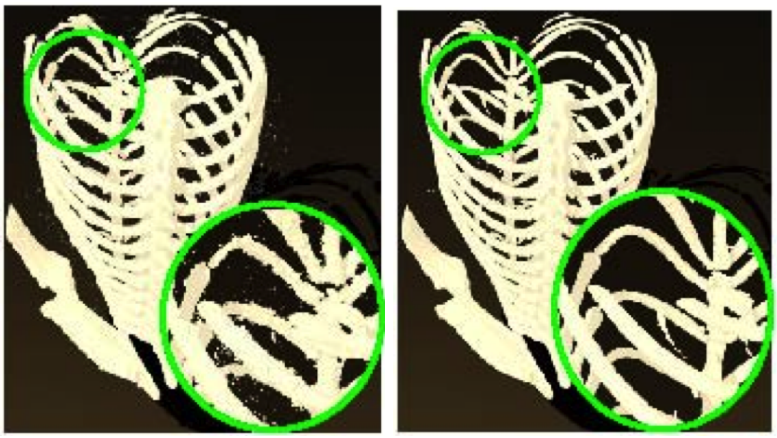
$$f_i(\mathbf{x}) = I(\mathbf{x}) - \mu_i(\mathbf{x}) \tag{4}$$

where $I(\mathbf{x})$ is the intensity value of the CT-scan at voxel $(\mathbf{x})$ and $\mu_i(\mathbf{x})$ is the average value of each region. This functional is supplemented by a regularization term involving the area and the volume of the surface. Minimizing the cost function in (3) w.r.t. ϕ_1 and ϕ_2 produces parabolic-type partial differential equations. The numerical solution of these equations with standard methods is slow, due to the stringent $\Delta t = O(\Delta x^2)$ time step restriction. In [3], the authors pointed out the connection between standard k-means algorithms and Chan-Vese algorithms [2, 15]. They then designed a hybrid algorithm taking advantage of both the efficiency of k-means and the regularization property of surface evolutions. In the present work, we use the regularization introduced in Bertelli et al. [1]. A distinguishing feature of the regularization of Bertelli et al. is that (1) the effect of the regularization does not separate two overlapping contours and (2) two contours running close to each other either snap onto each other or move apart from each other. These two properties are not enforced by other regularization procedures and does matter tremendously in the case of the segmentation of several regions. For example, figure 3 illustrates the artifacts produced by standard regularization procedures and the ability to accurately segment data into four phases with our approach. Overall, the combination of the extension of [3] with the regularization procedure of [1] produces an efficient and accurate segmentation into four phases algorithm as illustrated in figure 4.

4. **Spatial Representation and Refinement Criterion .** The computational cost for constructing the DRR is proportional to the number of cells visited. It is therefore desirable to lower the total number of cells used. On the other hand, it is necessary to accurately describe the regions where the attenuation factor varies rapidly, i.e. at the interface between phases such as the interface between bones and muscles or muscles and fat. An adaptive structure is thus highly desirable. We use a standard Octree data structure [9, 10] to efficiently represent the spatial discretization of the physical domain: Initially, the root of the tree is associated with the entire domain, then we recursively split each cell into eight children until the desired level of details is achieved. In this work, we choose to impose that the finest cells lie on the interface between regions, e.g. between bones and muscles. In addition, we do not impose that the ratio between cells be constrained. We use the criterion of [6, 5, 13] to automatically generate such grids, i.e. we split a cell C if:

$$\min_{v \in \text{vertices}(C)} |\phi(v)| \leq \text{Lip}(\phi) \cdot \text{diag-size}(C), \tag{5}$$

where diag-size(C) refers to the length of the diagonal of the cell, Lip(ϕ) is the Lipschitz constant and v refers to a vertex (node) of the current cell. Figure 5



(a) Standard regularization (b) Regularization procedure of [1]

FIGURE 3. Effects of the regularization process used in the segmentation algorithm of section 3, illustrating the successful removal of spurious noise.

(a) Rib cage (b) Bones-muscle-fat-air

FIGURE 4. Three dimensional boundaries detected by the segmentation algorithm described in section 3.

depicts a two dimensional projection of the Octree grid, superimposed to a section of the segmented CT-scan.

4.1. **Volume Representation via Octree .** In order to represent the CT volumetric data in a compact way using an Octree data structure and at the same time preserving details about the boundaries between different region, we need a measure of how distant each voxel is to the closest boundary. This will be used in the estimation of $\mathrm{Lip}(\phi)$ in 5. In fact a voxel close to the boundary indicate the need for a precise description of the volume in terms of very fine cells, whereas if voxel farther away from the boundary can be described using coarser cells.

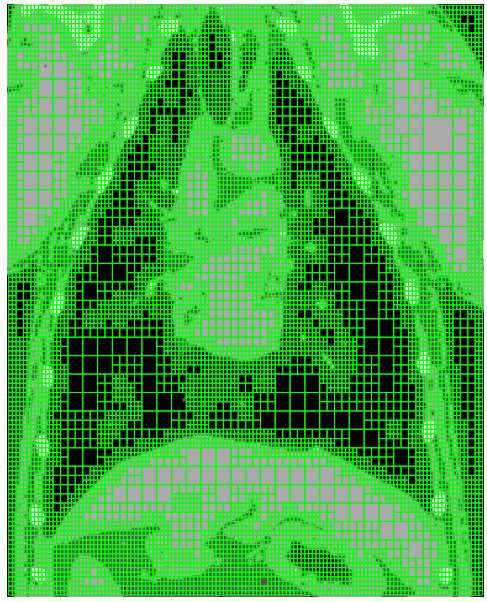


FIGURE 5. Example of a two dimensional cross-section of the three dimensional Octree grid obtained with algorithm 1, superimposed onto the segmentation map. In practice, higher ratios between the finest and coarsest cells can be considered.

In order to compute these distances we first 'reinitialize' the two level set functions ϕ_1 and ϕ_2, such that they become signed distance function to their corresponding boundaries. This is efficiently carried out using the Fast Marching Method (FMM) [14, 11], which computes the distances by solving the Eikonal equation:

$$|\nabla D(\mathbf{x})| = 1, \tag{6}$$

with complexity $O(n \log n)$, where n is the number of voxels. The initial conditions for (6) are chosen such that $D(\mathbf{x}) = 0$ for all the voxels $\mathbf{x}$ belonging to the interfaces computed by the segmentation algorithm. Once we performed the reinitialization of both ϕ_1 and ϕ_2, we have two signed distance functions D_1 and D_2. D_1 will encode information about distances from certain boundaries, while D_2 will encode the distances to the remaining boundaries. To combine the information of the two distance functions we create:

$$D(\mathbf{x}) = \min(|D_1(\mathbf{x})|, |D_2(\mathbf{x})|).$$

Once the distance map D is constructed, we initialize a uniform grid Octree G^n and we generate the adaptive grid Octree G^{n+1} by splitting or merging the cells of G^n according to Algorithm 1.

The parameter α, which appears in line 3 of Algorithm 1 can be thought of being an approximation of $\mathrm{Lip}(\phi)$ in equation (5) and controls the grading of the octree grid. For instance if $\alpha > 1$ the grid is built in a conservative way using mostly fine cells. On the other hand decreasing α, we can increase the size of the cells far away from the boundaries, obtaining a more compact representation of the three dimensional structure. Figure 5 illustrates a two dimensional projection of the Octree grid, superimposed to a section of the segmented CT-scan.

Algorithm 1 Octree Grid Generation

Input G^n and T_T

1: $G^{n+1} = G^n$
2: $C =$ root of G^{n+1}
3: **if** $\min_{v \in \text{vertices}(C)} |D(v)| \leq \alpha \cdot \text{diag-size}(C)$ **then**
4: **if** C is leaf **then**
5: split C
6: **end if**
7: **for** each child C' of C **do**
8: **go to** 3 with $C = C'$
9: **end for**
10: **else**
11: merge C
12: **end if**

return(G^{n+1})

5. **Construction of the Digital Reconstructed Radiograph .** The Digital Reconstructed Radiograph is computed by choosing a point in the three dimensional space as the virtual source of X-rays and by computing the attenuation of those rays passing through the CT volume and recording its value onto the DRR image plane. In fact, each voxel in the CT data has a particular attenuation value (depending on wether it is in the bone, muscle, fat or air regions), which can be considered directly proportional to the CT-scan intensity value. Standard techniques to generate DRRs via ray casting have a computational complexity $O(n^3)$, where n is the size of one side of the DRR (which we assume almost equivalent to each of the sides of the CT volume). In our case we can exploit the description of the CT volume obtained via adaptive grids to significantly reduce the computational complexity.

The entire ray casting procedure boils down to determining if a particular ray intersects a given three dimensional cell of the octree. If there is no intersection the entire branch of the tree that originates from that cell can be ignored. In the case there is an intersection, the children are recursively inspected. If a cell is a leaf and it intersects the ray, its attenuation coefficient (previously pre-computed) is added to the total attenuation index of that particular ray. The recursive algorithm is summarized in Algorithm 2.

To compute the intersection of a ray with an Axis Aligned Bounding Box (i.e. each cube of the octree structure) we used the state-of-the-art method proposed by Williams et al. in [16], which extends that proposed by Smits in [12]. The function *ComputeIntersection* in Algorithm 2, has been implemented following [16] and returns the attenuation index of the cell multiplied by the length of the intersection, in case there is an intersection, or a negative index, otherwise. Figure 1 depicts one example of Digital Reconstructed Radiograph obtained using the algorithm detailed in this section.

6. **Registration of the Fluoroscopy Image to the DRR.** The last step is the actual registration of the 3D CT-scan onto the X-ray fluoroscopy image (using the DDR). We follow the work of [17] and use a mutual information based method in which the rigid registration parameters of the pose are estimated. The method based on Mutual Information is very suitable to register image pairs produced from

Algorithm 2 Ray Casting Procedure through the Octree Grid

Main Loop

Input: Octree G^n and Ray r, Output: λ (attenuation)

```
C = root of G^n and λ = 0
Call InspectCell(C, λ, r)
return(λ)
```

Function InspectCell(C, λ, r)

```
λ_t = ComputeIntersection(C, r)
if λ_t > 0 then
    if C == leaf then
        λ = λ + λ_t
    else
        for each child C' of C do
            InspectCell(C', λ, r)
        end for
    end if
end if
```

different sensors modalities (such as in our case, DRR vs X-ray Fluoroscopy images); it aims at maximizing the amount of information that each pixel carries about the corresponding pixel in the registration. If the images are lined up this information is maximized. The proposed technique can be outlined as follows:

1. Initialize the rigid transformation $T = T_0$, which defines the pose. Here T, describing the pose, is a rigid transformation parameterized in terms of rotation, translation and scale (RST) with a total of 7 degrees of freedom.
2. Compute the DRRs as described in sections 3 through 5.
3. Compute the functional that has to be maximized (i.e. the mutual information $I\big(U(X), V(T(X))\big)$, where $U(X)$ is the 2-D fluoroscope image and $V(T(X))$ is the DRR generated for the pose T). Note that the value of the functional is not needed, since we only need its gradient to update the pose parameters in the direction of the gradient. The mutual information I can be expanded as:
$$\begin{aligned} I\big(U(X), V(T(X))\big) &= H(U(X)) + H(V(T(X))) \\ &- H(U(X), V(T(X))), \end{aligned}$$
where $H(A) = E_A(\log p(A))$ is the entropy of the random variable A. From here the gradient w.r.t. the parameters of T is computed.
4. Compute the new pose parameters using the updates computed in the previous step.
5. Go back to 2 and loop until convergence, i.e the L^∞ difference between two successive iterations is less than a threshold value.

Figure 6 depicts a sequence of results obtained with this algorithm demonstrating that we can successfully register 3D CT-scan with 2D X-ray fluoroscopy scans using the DRR.

7. **Conclusion.** We have presented a algorithm for the registration of 3D CT-scans to 2D X-ray fluoroscopy scans. In particular, we have introduced a novel approach

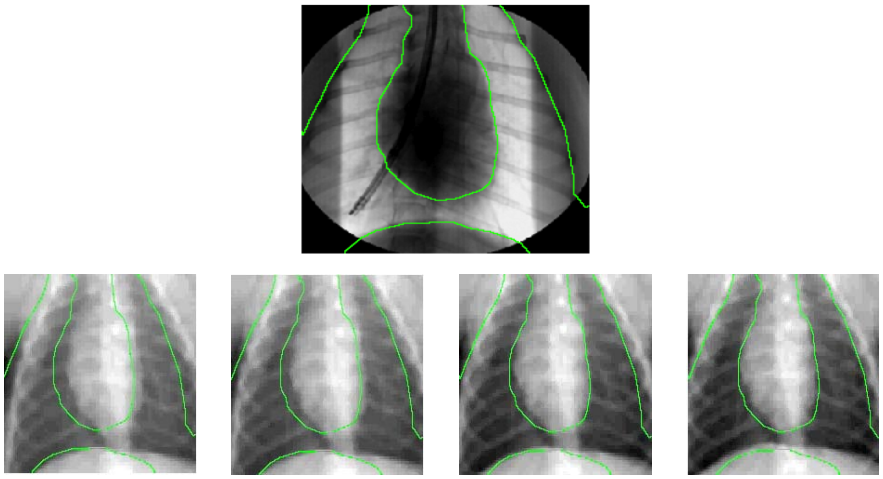


FIGURE 6. Top: 2D X-ray fluoroscopy on which we have outline the principal boundaries. Bottom: Sequence of images obtained with the registration procedure.

for the construction of the digital reconstructed radiographs needed for comparing two dimensional objects. Our approach takes advantage of an efficient representation of the main features in the CT-scan using an Octree data structure. In particular, the Octree construction is based on the location of the main anatomical features inferred from the segmentation of the CT-scan into four regions (bone, muscle, fat, air). The significant reduction in the number of voxel representing the CT-scan, while at the same time the accurate description of anatomical features, produces an efficient and accurate algorithm for the construction of the DDR. We have applied this framework to the registration of 3D CT-scans to 2D X-ray fluoroscopy scans following the work of [17].

Acknowledgment. The research of F. Gibou was supported in part by ONR under grant agreement N00014-11-1-0027, by the National Science Foundation under grant agreement CHE 1027817, by the Department of Energy under grant agreement DE-FG02-08ER15991, by the Institute for Collaborative Biotechnologies through contract no. W911NF-09-D-0001 from the U.S. Army Research Office and by the W.M. Keck Foundation.

REFERENCES

[1] L. Bertelli, S. Chandrasekaran, F. Gibou, and B. Manjunath. On the length and area regularization for multiphase level set segmentation. *International Journal on Computer Vision*, 90(3):267–282, 2010.

[2] T. F. Chan and L. A. Vese. Active contours without edges. *IEEE Transactions on Image Processing*, 10(2):266–77, February 2001.

[3] F. Gibou and R. Fedkiw. Fast hybrid k-means level set algorithm for segmentation. Technical report, Stanford, 2002. Also in the proceeding of the 4th International Conf. on Stat., Math. and Related Fields, Honolulu (2005).

[4] M. Levoy and P. Hanrahan. Light field rendering. In *Computer Graphics SIGGRAPH*, volume 30, pages 31–42, 1996.

[5] C. Min. Local level set method in high dimension and codimension. *J. Comput. Phys.*, 200:368–382, 2004.

[6] C. Min and F. Gibou. A second order accurate level set method on non-graded adaptive Cartesian grids. *J. Comput. Phys.*, 225:300–321, 2007.

[7] S. Osher and J. A. Sethian. Fronts propagating with curvature-dependent speed: Algorithms based on hamilton-jacobi formulations. *Journal of Computational Physics*, 79:12–49, 1988.

[8] D. Russakoff, T. Rohlfing, and C. R. Maurer. Fast intensity-based 2d-3d image registration of clinical data using light fields. In *IEEE International Conference on Computer Vision*, volume 1, pages 416–422, 2003.

[9] H. Samet. *The Design and Analysis of Spatial Data Structures*. Addison-Wesley, New York, 1989.

[10] H. Samet. *Applications of Spatial Data Structures: Computer Graphics, Image Processing and GIS*. Addison-Wesley, New York, 1990.

[11] J. Sethian. A fast marching level set method for monotonically advancing fronts. *Proc. Natl. Acad. Sci.*, 93:1591–1595, 1996.

[12] B. Smits. Efficient bounding box intersection. *Ray tracing news*, 15(1), 2002.

[13] J. Strain. Tree methods for moving interfaces. *J. Comput. Phys.*, 151:616–648, 1999.

[14] J. Tsitsiklis. Efficient algorithms for globally optimal trajectories. *IEEE Trans. on Automatic Control*, 40:1528–1538, 1995.

[15] L. A. Vese and T. F. Chan. A multiphase level set framework for image segmentation using the mumford and shah model. *International Journal of Computer Vision*, pages 271–293, December 2002.

[16] A. Williams, S. Barrus, R. K. Morley, and P. Shirley. An efficient and robust ray-box intersection algorithm. In *International Conference on Computer Graphics and Interactive Techniques*, 2005.

[17] L. Zollei, E. Grimson, A. Norbash, and W. Wells. 2d-3d rigid registration of x-ray fluoroscopy and ct images using mutual information and sparsely sampled histogram estimators. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 696–673, 2001.

Received April 13, 2011; Accepted May 14, 2012.

E-mail address: `lbertelli@ece.ucsb.edu`
E-mail address: `fgibou@engineering.ucsb.edu`