HAL
open science

An algorithm for posturing computational phantoms

Riccardo Scorretti

To cite this version:

Riccardo Scorretti. An algorithm for posturing computational phantoms. 1st EMF-Med, Sep 2018, Split, Croatia. 10.23919/EMF-MED.2018.8526079 . hal-01935824

HAL Id: hal-01935824

https://hal.science/hal-01935824v1

Submitted on 27 Nov 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

An algorithm for posturing computational phantoms

Riccardo Scorretti
[1] Université de Lyon, Ampère, UMR 5005, F-69134 Ecully Cedex, France, riccardo.scorretti@ec-lyon.fr

***Abstract*— Generation of postured computational phantom is crucial for simulating exposure to electromagnetic field in realistic situations. Most posturing algorithms are based on free form deformations (FFD), which have the defect of incorrectly deform rigid structures of the body (bones) unless specific precautions are taken. We propose an algorithm which guarantees that rigid structures are not deformed. Moreover, parametrization of the algorithm is much simpler than by using FFD.**

Keywords—numerical dosimetry, computational phantoms

I. Introduction

In order to simulate the exposure of workers to environmental electromagnetic fields, it is mandatory to take in to account the posture of the body. Unfortunately all computational phantoms are provided in a rigid, upright position. Therefore it is necessary to deform original phantoms in order to place them in the required posture. In the past some authors developed manually phantoms with some fixed postures [1]. However, this requires a cumbersome amount of work, for a result which is not generalizable to arbitrary postures. Nagaoka proposed to use free-form deformations to deform Japanese phantoms to arbitrary postures [2]. The same idea has been taken on more recently by Gao [3] and Nikolovski [4]. This approach requires to define a set of non-overlapping control lattices, the boundaries of which must match perfectly. The deformation is obtained by modifying lattices in a non-trivial way. A completely different strategy is adopted by Cherubini *et al.* [5] and Finet [6]. All voxels of the phantom are under the control of one or more volumes of influence. Each volume of influence is associated with a (rigid) bone. For each voxel, its coordinates with respect of each volume of influence are computed. Then, an interpolation between all these coordinates is performed in order to compute the coordinates of the voxel in the postured phantom. Several legacy (POSER, SIM4LIFE) and free software (BENDER) implement some of these algorithms. Algorithms may vary between these software, and unfortunately implementation details are seldom provided. In this preliminary work we propose a variant of algorithms presented in [5,6] where weight functions (which will be described below) are computed by solving a Laplace problem.

II. Methods

A. Skinning algorithm

Let $\mathbf{x}$ the coordinates of a generic voxel of the reference (not postured) phantom. The purpose of the algorithm is to compute the coordinates $\mathbf{y}$ of each voxel in the postured configuration (skinning). The main idea is that the positions of bones define the posture of the phantom. Each bone (or group of bones) of the body is associated with a unique volume of influence Ω_k. Volumes Ω_k are connected, and must contain the bone(s) to which they are associated. When the phantom is postured, bones and associated volumes eventually undergo the same rigid transformation. Conversely to algorithms based on FFD, volumes of influence[1] *must* overlap each other, so that each voxel is under the control of at least a volume of influence. That is:

$$\bigcup_{k=1\ldots N} \Omega_k = \Omega \tag{1}$$

where Ω indicates the whole body, and N is the number of volumes. Until this condition holds, there is a large freedom in the choice of each volume of influence. One could event take $\Omega_k = \Omega$, even if this is definitely not the best choice from the computational point of view. A coordinate system is associated with each volume Ω_k, so that for each voxel we can compute a set of local coordinates $\mathbf{u}_k$. In the reference configuration one has:

$$\mathbf{x} = \varphi_1 \mathbf{u}_1 = \varphi_2 \mathbf{u}_2 = \ldots = \varphi_N \mathbf{u}_N \tag{2}$$

where $\varphi_k : \mathbf{u} \mapsto \mathbf{x}$ is the map takes local coordinates $\mathbf{u}$ to global coordinates $\mathbf{x}$. Things change when the phantom is postured, because (2) does not hold any more. The idea is to compute the coordinates in the postured configuration as:

$$\mathbf{y} = \sum_{k=1\ldots N} w_k \cdot (\varphi_k \mathbf{u}_k) \tag{3}$$

where $w_k : \mathbf{x} \to [0\,;1]$ are weight functions which depend on the particular phantom; without loss of generality, we assume that for each voxel one has: $\sum_{k=1\ldots N} w_k = 1$. The coordinate of each voxel in the postured configuration is linearly interpolated from the set of coordinates $\{\varphi_k \mathbf{u_k}\}_{k=1\ldots N}$ which refer to all volumes. This method is called linear blending skinning [7]. Condition (1) guarantees that each voxel is under the control of at least a volumes. Voxels which are under the control of a single volume (namely Ω_i) move in a rigid way together with it:

$$w_k = \delta_{k,i} \qquad \Rightarrow \qquad \mathbf{y} = \sum_{k=1\ldots N} w_k \cdot (\varphi_k \mathbf{u}_k) = \varphi_i \mathbf{u}_i$$

[1] In FFD-based algorithms the phantom must be embedded in a control lattice, which defines a partition of the whole body. The different parts of the control lattice must not overlap, and must match perfectly on their common boundaries.

The case of voxels which are under the control of several volumes is more complicated. The expected behavior is defined by the following conditions: 1) bones moves in a rigid way with its own volume of influence. Conversely, other volumes have strictly no influence on these voxels. 2) Weight functions must be continuous and vanish on boundaries, so that the map $\mathbf{x} \mapsto \mathbf{y}$ defined by (3) is also continuous. 3) The more voxels are close to a bone, the more they "feel" the influence of that bone. One sees that these conditions are all satisfied if weight functions are the solution of the following Laplace problems:

$$\begin{cases} \Delta w_k = 0 \\ w_k = 0 \text{ on } \partial\Omega_k \text{ and in void voxels} \\ w_k = \delta_{k,i} \text{ for bones associated with } \Omega_i \end{cases} \quad (4)$$

B. Computation of weight functions

Eq. (4) must be solved for each volume of influence Ω_k. Even if this calculation is computationally intensive, it has to be observed that it is a one-time task. Moreover, as a high precision is not required to solve (4) for each volume, so that computational time can be reduced. We discretize (4) by Finite Differences. A first solution is computed on a coarse grid (10^5 unknown) by using LU factorization as linear solver. This coarse solution is then projected on the fine grid (10^7 unknowns) and a few Jacobi iterations are executed in order to smooth the solution. The relative precision which is obtained is of the order of 10^{-2}. We tried to use GMRES as linear solver but results were deceiving: even if the precision was higher, negative values could be obtained for $w_k(\mathbf{x})$, so we finally prefer to use Jacobi method, even if it is slower. By using this strategy, computation can be performed on any standard PC in a reasonable time (about 2-3 minute for each volume).

C. Comparison with existing algorithms

The proposed algorithm is extremely similar to [5], which is implemented in the software SIM4LIFE. However in this reference the method of computation of $w_k(\mathbf{x})$ is not described in detail, so it is not possible to compare further the two algorithms. The same remark applies to [6], which is implemented in the software BENDER: it is generically stated that by default functions w_k (called skinning maps) are *"generated by a heat diffusion algorithm"*.

III. CONCLUSION

A preliminary example of application of our algorithm with phantom Duke is depicted Fig. 1. It can be observed that the phantom is deformed in a visually realistic way. Continuity of blood vessels is ensured, and bones don't deform, neither intersect. Thanks to the boundary conditions imposed in (4), no manual editing has been required for this example, even in the case of a complicate junction like shoulder. However, one should keep in mind that this algorithm still is an approximation: in particular, the choice of using Laplace operator in (4) is purely heuristic: solving a large-deformation elasticity problem would be more appropriate – but also so much more complicated. The definition of volumes of influence is somehow arbitrary. We tested our algorithm with volumes of different sizes, and the obtained results were quite similar, provided that there is enough overlap between adjacent volumes. Future work will be devoted to more extensive tests of our algorithm, in particular to quantify the variation of volume of different compartments of the body, and to check the effectiveness with other structures of the body. Another point which deserves investigation is the generation of tetrahedral mesh of the postured phantom without passing through the voxelization step.

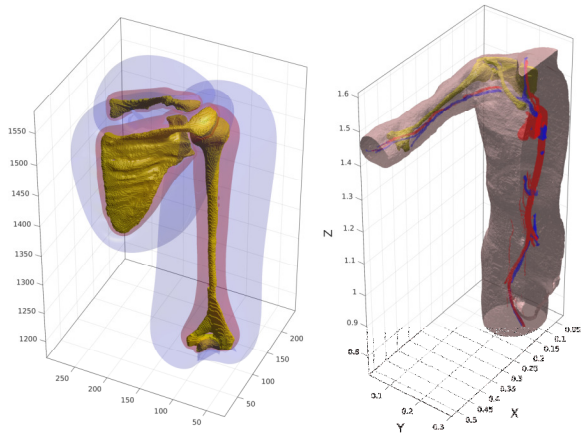

Fig. 1. Left: isosurfaces of weight functions corresponding to clavicle, scapula and homerus. Right: phantom (Duke) postured with right arm rotated of 90° (detail).

ACKNOWLEDGMENT

This work is been supported in part by The French National Research Program for Environmental and Occupational Health of Anses (2015/2 RF/02) and by PRIMES Labex ANR-11-LABX-0063 / ANR-11-IDEX-0007 grants.

REFERENCES

[1] T. W. Dawson, K. Caputa, and M. A. Stuchly, "Numerical evaluation of 60 Hz magnetic induction in the human body in complex occupational environments", Physics in Medicine and Biology, vol. 44, nº 4, p. 1025‑1040, avr. 1999.

[2] Nagaoka and S. Watanabe, "Postured voxel-based human models for electromagnetic dosimetry", Physics in Medicine and Biology, vol. 53, nº 24, p. 7047‑7061, déc. 2008.

[3] J. Gao, "Generation of postured voxel-based human models used for electromagnetic applications", Darmstadt, 2012.

[4] M. S. M. Nikolovski, "Detailed Modeling of the Human Body in Motion to Investigate the Electromagnetic Influence of Fields in a Realistic Environment", Darmstadt, 2017.

[5] E. Cherubini, N. Chavannes, and N. Kuster, "Realistic Skeleton Based Deformation of High-Resolution Anatomical Human Models for Electromagnetic Simulations", The 31st Annual Meeting of the Bioelectromagnetic Society, p. 88.

[6] J. Finet, R. Ortiz, J. Andruejol, A. Enquobahrie, J. Jomier, J. Payne and S. Aylward, "Bender: An Open Source Software for Efficient Model Posing and Morphing". In International Symposium on Biomedical Simulation (pp. 203-210). Springer, Cham.

[7] L. Kavan and J. Žára, "Spherical blend skinning: a real-time deformation of articulated models", Proceedings of the 2005 symposium on Interactive 3D graphics and games (pp. 9-16). ACM.