Medical Image Segmentation and the Use of Geometric Algebras in Medical Applications

Rafael Orozco-Aguirre, Jorge Rivera-Rovelo, and Eduardo Bayro-Corrochano

CINVESTAV, Unidad Guadalajara,
López Mateos Sur 590, Zapopan, Jalisco, México
{horozco, rivera, edb}@gdl.cinvestav.mx

Abstract. This paper presents a method for segmentation of medical images and the application of the so called geometric or Clifford algebras for volume representation, non-rigid registration of volumes and object tracking. Segmentation is done combining texture and boundary information in a region growing strategy obtaining good results. To model 2D surfaces and 3D volumetric data we present a new approach based on marching cubes idea however using spheres. We compare our approach with other method based on the delaunay tetrahedrization. The results show that our proposed approach reduces considerably the number of spheres. Also we show how to do non-rigid registration of two volumetric data represented as sets of spheres using 5-dimensional vectors in conformal geometric algebra. Finally we show the application of geometric algebras to track surgical devices in real time.

1 Introduction

When dealing with tumor segmentation in brain images, one way to solve the problem is by using Magnetic Resonance (MR) images because in such images we have different types of them (ie. T1, T2, T1-weighted, T2-weighted, etc.; some of them highlight tumor and other structures), and by combining and differentiating them, the task become more easy and an automatic approach for segmentation become possible (see [1]). Other methods, like the one proposed by [2], use a probabilistic digital brain atlas to search abnormalities (outliers) between the patient data and the atlas. The use of Computer Tomographic (CT) images is less used because they have not such modalities and the development of an automatic algorithm for segmentation is more complicated; however semi-automatic approaches have been proposed (as in [3,4]) using seed points defined manually by the user as initialization, and growing the region by some method. In this work we are interested in segmenting tumors in CT images, so we use a simple but effective algorithm to segment them: a set of 5 *texture descriptors* is used to characterize each pixel of the image by means of 5×1 template or a 5D-vector; then each vector is compared with the typical vector describing a tumor in order to establish an initialization of the tumor in the image (seed points for tumor tissue). Finally, a region growing strategy is used, combined

with boundary information to obtain the final shape of the tumor (this method is explained in section 2).

On the other hand, representation of volumetric objects using primitives like points, lines or planes is a common task. The Union of Spheres proposed in [5] is another possible representation for volumetric data, but it usually needs a large amount of primitives (spheres). This fact aimed us to look a different way to model the object with less primitives but being a good enough representation. In the first approach, the dense Union of Spheres representation is obtained using the Delaunay tetrahedrization and its complexity is $O(n^2)$ in both, time and number of primitives, while our highest number of spheres using our method based on marching cubes is less than $2n$ in the worst case, and some times it is less. We use computer tomography (CT) images to do the experiments, and one of the the surfaces to be modeled is the segmented tumor - n is the number of boundary points in a total of m CT images (slides). This approach is explained in section 4, which uses the concepts explained in section 3.

Some times (ie., when surgeon opens the head and occurs loss of cerebrospinal liquid) tumor and brain structures suffer (non-linear) deformation. In this work (see section 4.2) we present a new approach which uses models based on spheres for using such spheres as the entities to be aligned. This is embedded in the Conformal Geometric Algebra (CGA) framework using the TPS-RPM algorithm but in a 5-dimensional space (see Sect. 4.2). Finally, we show the application of GA for the task of object tracking (section 5).

2   Segmentation

As mentioned in [7,8] segmentation techniques can be categorized in three classes: a) thresholding, b) region-based and c) boundary-based. Due to the advantages and disadvantages of each technique, many segmentation methods are based on the integration information of region and boundary techniques and there are a great variety of methods; some of them working better in some cases, some being more sensitive to noise, etc. This fact make not feasible to determine the best approach to segmentation that integrates boundary and region information because we have not a generally accepted and clear methodology for evaluating the algorithms; additionally, the properties and objectives that the algorithms try to satisfy and the image domain in which they work are different. Interested reader can consult a detailed review of different approaches in [7]. Due to the fact that we are dealing with medical images, we need also to take into account an important characteristic: the texture. Textural properties of the image can be extracted using *texture descriptors* which describe the texture in an area of the image. So, if we use a *texture descriptor* over the whole image, we obtain a new "texture feature image". In most cases, a single operator does not provide enough information about texture, and a set of operators need to be used. This results in a set of "texture feature images" that jointly describe the texture around each pixel.

When segmenting tomographic images, simple segmentation techniques such as region growing, split and merge or boundary segmentation can not be used

alone due to the complexity of the brain computer tomographic images, which contain textures of different tissues, similar gray-levels between healthy and non-healthy tissues, and sometimes the boundaries are not well defined. For this reason, we decide to combine not only boundary and region information (as typically it is done), but also to integrate information obtained from texture descriptors and embed that in a region growing strategy. A block diagram of our approach is shown in figure 1.a.
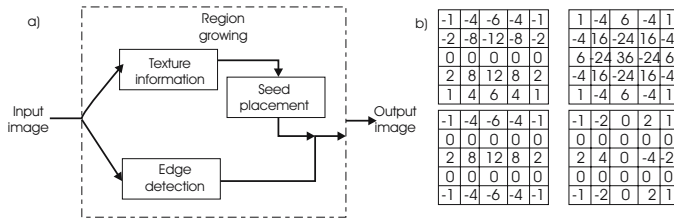
Fig. 1. a) Block diagram of the approach to segment tumors in CT images (region growing strategy combining texture and boundary information); b) Texture descriptors used to obtain the texture information (4 Laws energy masks)

The first step is to characterize each pixel on images, so we opt for use the texture information provided by some of the Laws's masks to characterize them with a five-dimensional vector (named *texture vector*, V_{ij}, for pixel in coordinates (i, j)). Then, to place automatically the seed points for the region growing strategy, we choose only the pixels having a texture vector for the tissue of interest (in this case we are interested in tumor) and use them as initialization (or seeds) for the region growing strategy; boundary information is used to stop the growing of the region. The construction of V_{ij} is explained as follows: the first element of V_{ij} is only to identify if the pixels corresponds to the background (value set to zero) or to the patient's head (value set to one) - patient's head could be skin, bone, brain, etc.; in order to obtain the texture information, we use a set of four masks of the so called Laws Masks (L5E5, R5R5, L5S5, E5S5 - see 1.b); then we fix the value in a position of V_{ij} with 1's or 0's, depending on if the value is greater than zero or zero, respectively. As a result, each structure (tissue, bone, skin, background) on the medical images used, has the same vector V_{ij} in a high number of its belonging pixels, but not in all of them because of variations in values of neighboring pixels. So we can use the pixels having the texture vector of the object we want to extract to establish them as seed points in a region-growing scheme. Region growing criterions we use are as follows: we compute the mean μ_{seeds} and standard deviation σ_{seeds} of the pixels fixed as seeds; then, for each neighboring pixel being examined to determine if added or not to the region:

If $I(x, y) = \pm 2\sigma_{seeds}$ and $V_{xy} \neq V_{seed}$ at most in 1 element, then $I(x, y) \in R_t$

where R_t is the region of the tumor. The stopping criterion takes into account the boundaries of the object because the growing of the region is in all directions, but when a boundary pixel is found, the growing in such direction is stopped. Figure 2 shows results of the process explained before: figure 2.a shows one original CT-image; figure 2.b shows the seed points fixed, which have the texture vector of the tumor; figure 2.c shows the final result after the overall process has ended (the tumor extracted). The overall process takes only few seconds per image and it could be used to segment any of the objects; but in our case, we focus our attention on the extraction of the tumor.
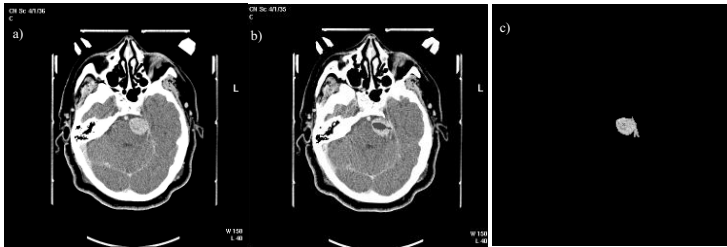

Fig. 2. Results for the segmentation. a) One of the original CT-images; b) Seed points fixed; c) Result for the image of (a) after the whole process (the tumor extracted).

After that, the next step is to model the volumetric data by some method. Due to the fact that tumor can be deformed due to the lost of cefalic liquid once the head of the patient is opened, we need a 3D representation of the tumor which allows us to estimate such deformation to update the shape of the tumor. Next sections explain the basis of our different approach for such modeling as well as a similar method used for comparison. However, first we present how the spheres are represented in conformal geometric algebra (CGA), and then we will show how to build 3D models and register two of them using such entities with TPS-RPM method.

3   Representation of Spheres in CGA

Our objective is not to provide a detailed description of the geometric algebra (GA) and its advantages (interested reader can find very useful material in [10,11]), so we only give a brief introduction and explain how to represent spheres in conformal geometric algebra (CGA) as points in a space of 5 dimensions (because such representation will be used in the non-rigid registration process).

Geometric algebra is a coordinate-free approach to geometry based on the algebras of Grassmann and Clifford. The algebra is defined on a space whose elements are called *multivectors*; a multivector is a linear combination of objects of different grade, e.g. scalars, vectors and k-vectors. It has an associative and

fully invertible product called the *geometric* or Clifford product. The existence of such a product and the calculus associated with the geometric algebra endows the system with tremendous power. The Clifford product (or geometric product) ab between two vectors a and b is defined as:

$$ab = a \cdot b + a \wedge b \ . \tag{1}$$

where $a \cdot b$ represents the *dot* or *inner* product and $a \wedge b$ represents the *wedge* or *exterior* product. The geometric algebra $G_{p,q,r}$ is a linear space of dimension 2^n, where $n = p + q + r$ and p, q, r indicate the number of basis vectors which squares to $1, -1, 0$, respectively. This algebra is constructed by the application of geometric product between each two basis vectors e_i, e_j from the base of the vector space $\Re^{p,q,r}$. Thus $G_{p,q,r}$ has elements of grade 0 (scalars), grade 1 (vectors), grade 2 (bivectors), and so on. The CGA $G_{4,1,0}$ is adequate for representing entities like spheres because there is no direct way to describe them as compact entities in $G_{3,0,0}$ (the geometric algebra of the 3D space); the only possibility to define them is given by formulating a constraint equation. However, in CGA the spheres are the basis entities from which the other entities are derived. These basic entities, the spheres $\underline{s}$ with center p and radius ρ are defined by (2).

$$\underline{s} = p + \frac{1}{2} \left(p^2 - \rho^2 \right) e + e_0 \ . \tag{2}$$

where $p \in \Re^3$, ρ is a scalar and e, e_0 are defined as in eq. 3 (they are called null vectors), and they are formed with two basis vectors e_-, e_+ additional to the three basis vectors of the 3D-Euclidean space (which have the properties that $e_-^2 = -1; e_+^2 = +1; e_- \cdot e_+ = 0$).

$$e = e_- + e_+; \quad e_0 = \frac{1}{2}(e_- - e_+) \tag{3}$$

In fact, we can think in a conformal point $\underline{x}$ as a degenerate sphere of radius $\rho = 0$. More details on GA and the construction of other entities in CGA can be consulted in [10,11]. We can see eq. 2 as a linear combination: $\underline{s} = \alpha e_1 + \beta e_2 + \gamma e_3 + \delta e_+ + \epsilon e_-$, or represent it as a 5D-vector $\underline{s} = [\alpha \ \beta \ \gamma \ \delta \ \epsilon]^T$. Thus, the sphere in CGA is represented with a 5-dimensional vector, which is an adequate representation to make two sets of 5-vectors, one representing the object and the other the deformed object. These sets are obtained by the method explained in next section (4). Once we have these sets, we will be able to apply the TPS-RPM algorithm in order to do the registration process (see Sect. 4.2). However, let us explain before how the rigid motion is done in GA. In GA, rotations are computed by the so called *rotor*, R, defined as in equation 4, where a is the plane perpendicular to the rotation axis; while translations are computed by the *translator*, T, defined as in equation 5, where t is the translation vector and e is defined as in 3.

$$R = \exp{-\frac{1}{2}\theta a} \tag{4}$$

$$T = \exp{-\frac{t}{2}e} \tag{5}$$

To rotate any entity in any dimension, we multiply it by the rotor R from the left and by the conjugate $\tilde{R}$ from the right, $x' = Rx\tilde{R}$. Translations are made in the same way: $y' = Ty\tilde{T}$. If we combine the rotation and the translation, the resulting operator is named *motor* and is expresed as $M = TR$, which is applied in the same way explained: $x' = Mx\tilde{M} = TRx\tilde{R}\tilde{T}$.

4 Volume Representation and Non-rigid Registration

In medical image analysis, the availability of 3D-models is of great interest to medicians because it allows them to have a better understanding of the situation, and such models are relatively easy to build. However, in special situations (as surgical procedures), some structures (as brain or tumor) suffer a (non-rigid) transformation and the initial model must be corrected to reflect the actual shape of the object. For this reason, it is important to have a representation suitable to be deformed, with the minor quantity of primitives involved in such representation as possible to make faster the process. In literature we can find the Union of Spheres algorithm (see [5]), which uses the spheres to build 3D-models of objects and to align or transform it over time. Nevertheless, we use the marching cubes algorithm's ideas to develop an alternative method to build 3D models by using spheres, which has the advantage of reducing the number of primitives needed. For space reasons we do not provide an explanation of the Union of Spheres nor the Marching Cubes algorithms, but it can be found in [5,9].

4.1 3D Models Using Spheres

To build a 3D model of the object of interest using spheres, we are based in the marching cubes algorithm (MCA). The principle of our propposal is the same as in MCA: given a set of m slides (CT images), divide the space in logical cubes (each cube contains eight vertices, four of slide k and four of slide $k + 1$) and determine which vertices of each cube are inside (or on) and outside the surface. Then define the number of spheres of each cube according to figure 3 and eq. 6 (where i is the ith sphere of the case indicated by j), taking the indices of the cube's corners as the first cube of such figure indicates. Note that we use the same 15 basic cases of the marching cubes algorithm because the total of 256 cases can be obtained from this basis. Also note that instead of triangles we define spheres and that our goal is not to have a good render algorithm (as intended for Marching cubes algorithm), but have a representation of the volumetric data based on spheres which, as we said before, could be useful in the process of object registration.

$$s_{p_i}^j = c_{p_i} + 0.5(c_{p_i}^2 - \rho_{p_i}^2)e + e_0 \;\; ; \;\; s_{m_i}^j = c_{m_i} + 0.5(c_{m_i}^2 - \rho_{m_i}^2)e + e_0$$
$$s_{g_i}^j = c_{g_i} + 0.5(c_{g_i}^2 - \rho_{g_i}^2)e + e_0 \tag{6}$$

Table 1 is a comparison between the results of the Union of Spheres and our approach for the case of a brain model. The first row shows the worst case with
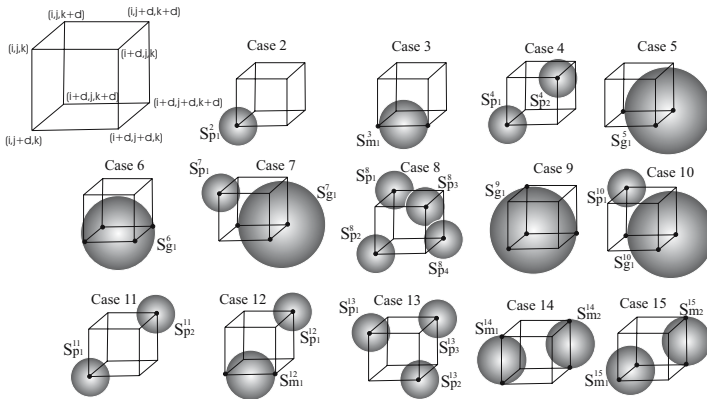
Fig. 3. The basic 15 cases of surface intersecting cubes (defining a different number of spheres with different centers and radius

both approaches; second row shows the number of spheres with improvements in both algorithms (reduction of spheres in DT is done by grouping spheres in a single one which contents the others, while such reduction is done using a displacement of $d = 3$ in our approach). The number of boundary points was $n = 3370$ in both cases. It is obvious the reduction in the number of primitives obtained with our approach, while maintaining clear enough the representation (even in the worst case). Figure 4.a-d shows the results obtained for a set of 36 images of a real patient with a tumor visible in 16 of them (see in figure 4.d the 3D model of the tumor of the real patient).

Table 1. Comparison between number of spheres using approach based on Delaunay tetraherization and our approach based on marching cubes algorithm; n is the number of boundary points; d is the distance between vertices in logical cubes of second approach.)

n/d	Num of spheres with each approach	
	DT approach	Our approach
3370 / 1	13480	11866
3370 / 3	8642	2602

4.2 Registration of Two Models

Suppose you have two points sets and one of them results from the transformation of the other but you do not know the transformation nor the correspondences between the points. In such situation you need an algorithm that find these two unknowns the best as possible. If in addition the transformation is non rigid, the complexity increases enormously. In the variety of registration algorithms
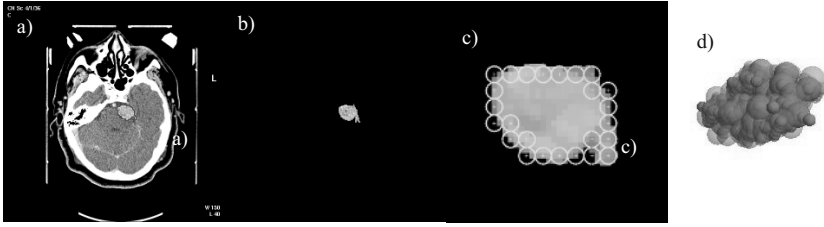
Fig. 4. Real patient: a) Original of one CT slide; b) Segmented object (the tumor); c) Zoom of the approximation by circles according the steps described in section; d) Approximation by spheres of the tumor extracted

existing today, we can find two that solve for correspondence and transformation: Iterated Closest Point (ICP) and Thin plate spline-Robust Point Matching (TPS-RPM). Details of each one of this algorithms can be found in [6]; here we assume, for space reasons, the reader knows them. In a past work we presented a comparison between these algorithms for non-rigid registration and we concluded TPS-RPM gives better results. However, we had used only sets of 2D and 3D points. Now we have spheres as points in a 5D-space modeling the object, and these spheres have not only different centers, but also different radius. So, for the non-rigid registration we follow the simulated annealing process of TPS-RPM explained in [6]. Let be $U_I = \{\underline{s}_j^I\}, j = 1, 2, ..., k$, the initial spheres set; $U_F = \{\underline{s}_i^F\}, i = 1, 2, ..., n$, the final spheres set. To update the matrix M of correspondence for spheres $\underline{s}_j^I$ y $\underline{s}_i^F$, modify m_{ji} as

$$m_{ji} = \frac{1}{T} e^{-\frac{(\underline{s}_i^F - f(\underline{s}_j^I))^\top (\underline{s}_i^F - f(\underline{s}_j^I))}{T}} . \tag{7}$$

for outlier entries $j = k + 1$ and $i = 1, 2, ..., n$:

$$m_{k+1,i} = \frac{1}{T_0} e^{-\frac{(\underline{s}_i^F - f(\underline{s}_{k+1}^I))^\top (\underline{s}_i^F - f(\underline{s}_{k+1}^I))}{T_0}} . \tag{8}$$

and for outliers entries $j = 1, 2, ..., k$ and $i = n + 1$:

$$m_{j,n+1} = \frac{1}{T_0} e^{-\frac{(\underline{s}_{n+1}^F - f(\underline{s}_j^I))^\top (\underline{s}_{n+1}^F - f(\underline{s}_j^I))}{T_0}} . \tag{9}$$

where T is the parameter of temperature which is reduced in each stage of the optimization process beginning at a value T_0 (remember that TPS-RPM use the simulated annealing process). Then, to update transformation we use the QR-decomposition of M to solve eq. 10 (following the same process explained in [6] and omitted here for space reasons).

$$E_{tps}(d, w) = \|Y - Vd - \Phi w\|^2 + \lambda_1(w^T \Phi w) + \lambda_2[d - I]^T[d - I] . \tag{10}$$

Figure 5.a shows the 3D models as sets of spheres representing the object (the tumor mentioned in figure 4) -one is the initial set (or representation at time t_1);
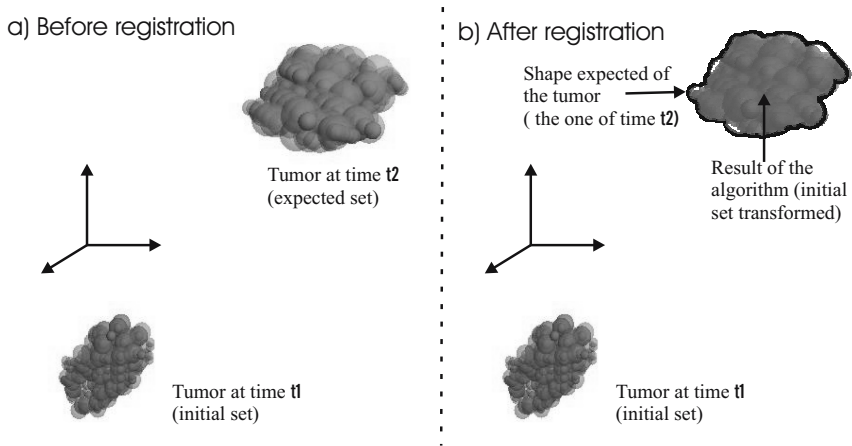
a) Before registration

Tumor at time t2
(expected set)

Tumor at time t1
(initial set)

b) After registration

Shape expected of
the tumor
(the one of time t2)

Result of the
algorithm (initial
set transformed)

Tumor at time t1
(initial set)

Fig. 5. a) Initial and expected sets (the expected set is obtained by a non-rigid trans-
formation of the initial one); b) Initial and result of applying TPS-RPM to align the
sets of spheres, represented as 5D-vectors in conformal geometric algebra. Note that
the resulting set has been aligned an looks like the initial one.

the other is the deformed or expected set (or representation at time t_2)- which must
be registered. Figure 5.b shows the results of registration process using TPS-RPM
algorithm with the spheres as 5D-vectors in conformal geometric algebra. Note that
usually, researchers use TPS-RPM with 2D or 3D vectors because they can not go
beyond such dimension; in contrast, using conformal geometric algebra we have an
homogeneous representation which preserves isometries and uses the sphere as the
basic entity. In figure 5, at the left are only the initial and expected sets; at the right
the initial and the result of registration but with the shape of the expected set for
visual comparison. Note that the algorithm adjusted the radius as expected (this
is not possible using only 3D vectors).

5   Object Tracking

Other important task in surgical procedures is the tracking of objects involved
in such procedures. For this purpose, some spherical markers are placed on the
instruments, and such markers are tracked using the Polaris System (Northern
Digital Inc.). To find the transfomation relating the 3D position of the objects
being tracked with the virtual model showed on display, we first calibrate the
real position of the patient with the 3D-model using the TPS-RPM algorithm
(section 4.2). Then we use the so called "motor" (explained in 3), to update the
position of the surgical devices in real time. The procedure to track is explained
as follows: first, we take two 3D point sets $\{\mathbf{x}_i\}$ and $\{\mathbf{x}_i'\}$ defined in the Euclidean
3D geometric algebra and compute the rotor $\mathbf{R}$ and the translation vector $\mathbf{t}$ which
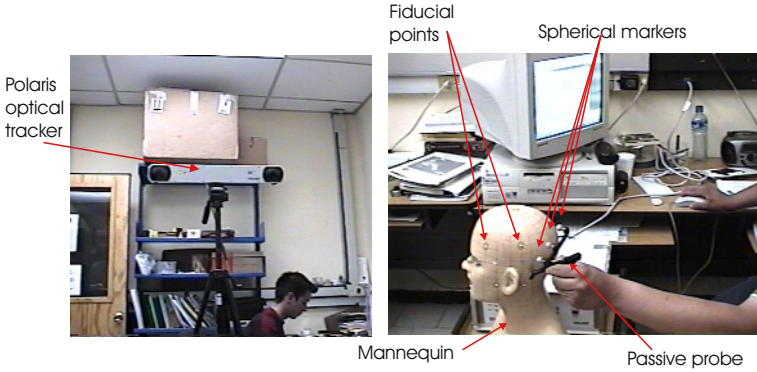minimize the following equation

Fig. 6. Scenario for tracking of devices: fiducial points are used to register the 3D model with what is been observed by the polaris system; spherical markers on device are used to track it

$$S = \sum_{i=1}^{n} \left[\mathbf{x}'_i - \mathbf{R}(\mathbf{x}_i - \mathbf{t})\tilde{\mathbf{R}} \right]^2. \tag{11}$$

The equations to compute the rotor and translation vector are obtained using the differentiations of equation (11).

$$F_{\alpha\beta} \equiv \sigma_\alpha \cdot \underline{f}(\sigma_\beta) = \sum_{i=1}^{n} (\sigma_\alpha \cdot \mathbf{u}_i)(\sigma_\beta \cdot \mathbf{v}_i) \tag{12}$$

$$\mathbf{t} = \frac{1}{n} \sum_{i=1}^{n} \left[\mathbf{x}_i - \tilde{\mathbf{R}}\mathbf{x}'_i\mathbf{R} \right] \tag{13}$$

where $\mathbf{u}_i = \mathbf{x}_i - \overline{\mathbf{x}}$ and $\mathbf{v}_i = \mathbf{x}'_i$. By computing the SVD of F we get $F = USV^T$ and using this result we compute the 3×3 rotation matrix $R = VU^T$. Thereafter the translation is computed using equation (13). This method was developed by Lasenby et al. in [12]. The exponential representation of the transformation in our framework reads

$$\mathbf{M} = \mathbf{R} + \frac{\mathbf{t}}{2}\mathbf{R} = e^{\mathbf{l}(\frac{\theta_u}{2} + e\frac{\mathbf{t}_u}{2})} \tag{14}$$

where θ_u is the angle and $\mathbf{t}_u$ the displacement with respect to the screw axis line $\mathbf{l}$. Applying this transformation to each point $\mathbf{x}'$, we can obtain a tracking path as follows:

$$\mathbf{x}' = \mathbf{TRx}\tilde{\mathbf{R}}\tilde{\mathbf{T}} = e^{\frac{1}{2}\mathbf{t}e}e^{\frac{\theta_u}{2}\mathbf{n}} \mathbf{x}_h \ e^{-\frac{\theta_u}{2}\mathbf{n}}e^{-\frac{1}{2}\mathbf{t}e}$$
$$= e^{\mathbf{l}(\frac{\theta_u}{2} + e\frac{\mathbf{t}_u}{2})} \mathbf{x}_h \ e^{-\mathbf{l}(\frac{\theta_u}{2} + e\frac{\mathbf{t}_u}{2})} \tag{15}$$

Figure 7 shows the applicaton of procedure explained before when tracking a "polaris in-line passive probe" with three spherical markers (as the one showed
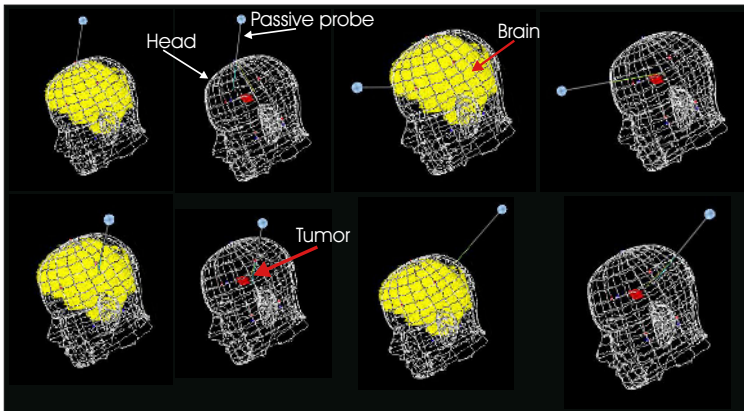
Fig. 7. Images in 3D virtual world of the process of tracking of the surgical device. First and third column: the whole 3D-model (skin + brain +device); second and fourth column: brain hidden to visualize the tumor.

in figure 6:b). The scenario is as follows (see figure 6):a mannequin (in substitution of a real patient); in such mannequin we put nine fiducial markers used to align the "presurgical 3D-model" with the real position when tracking is intended to be. A brain, obtained from a digital atlas, which is segmented and merged with the 3D-model of the mannequin in order to have a more realistic representation in the experiment, together with a tumor (also segmented to visualize in differents views). Figure 7 shows the 3D-model of the mannequin, brain, tumor and the device being tracked; such figure shows different momments while tracking the device. In such figure, the first and third column show the model complete (head+brain+device), the second and fourth one only the the head and the tumor for better visualization of the last one.

6   Conclusions

We have shown the application of GA in three different tasks: volume representation, non-rigid registration of sets of spheres and real time tracking. Also we show at the begining a different approach for medical image segmentation which combines texture and boundary information and embed it into a region-growing scheme, having the advantage of integrating all the information in a simple process. The algorithm proved to be very useful despite the limitations of the used CT images (limitations compared with the facilities given by MRI images, commonly used in similar works). With the GA framework, we show how to obtain a representation of volumetric data using spheres; our approach is based on the ideas exposed in marching cubes algorithm but it is not intended for rendering purposes or displaying in real time, but for reduce the number of primitives modeling the volumetric data and use less primitives in the process of registration. Also, we show how to represent these primitives as spheres in the

conformal geometric algebra, which are 5-dimensional vectors that can be used with the principles of TPS-RPM. Experimental results seem to be promising and highlight the potential of GA used in different tasks.

References

1. S. Ho, E. Bullitt, G. Gerig, "Level-set evolution with region competition: automatic 3-D segmentation of brain tumors", *Proceedings of 16th International Conference on Pattern Recognition*, Volume 1, pp. 532-535, 2002.
2. M. Prastawa, E. Bullit, S. Ho and G. Gerig, "A Brain Tumor Segmentation Framework Based on Outlier Detection," *Medical Image Analysis Journal*, 8(3), pp. 275-83, September 2004.
3. M.C. Andrade, "An interactive algorithm for image smoothing and segmentation," *Electronic Letters on Computer Vision and Image Analysis*, **4-1**, pp. 32-48, 2004.
4. P. Lin, C. Zheng, Y. Yang and J. Gu, "Medical Image Segmentation by Level Set Method Incorporating Region and Boundary Statistical Information," *9th Iberoamerican Congress on Pattern Recognition*, National Institute of Astrophysics, Optics and Electronics (INAOE), Puebla, Mexico, pp. 654-660, October 2004.
5. V. Ranjan and A. Fournier, "Union of spheres (UoS) model for volumetric data,", in *Proceedings of the Eleventh Annual Symposium on Computational Geometry*, Vancouver, Canada, 1995, C2-C3, pp. 402-403.
6. H. Chui, A. Rangarajan, "A new point matching algorithm for non-rigid registration". *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, Volume 2, pp. 44-51, 2000.
7. X. Muñoz, "Image segmentation integrating color, texture and boundary information," *Ph.D. Thesis in Computer engineering*, Girona, December 2002.
8. K.S. Fu and J.K. Mui, "A survey on image segmentation," *Pattern Recognition*, 12:395-403, 1980.
9. W. Lorensen and H. Cline, "Marching cubes: a high resolution 3D surface construction algorithm," *Computer Graphics.* **21-4**, 163–169, July 1987.
10. E. Bayro-Corrochano and G. Sobczyk, *Geometric algebra with applications in science and engineering*, Birkhuser, 2001.
11. B. Rosenhahn and G. Sommer, "Pose Estimation in Conformal Geometric Algebra," Technical report 0206, Christian-Albrechts-University of Kiel, November 2002, pp. 13-36.
12. J. Lasenby, A.N. Lasenby, C. Doran and W.J. Fitzgerald, "New geometric methods for computer vision - an application to structure and motion estimation," *International Journal of Computer Vision*, 26(3), pp. 191-213, 1998.