Model-based multiple object video tracking for treatment room supervision *

M.A. Portela Sotelo [†‡],
É. Desserée[†], J-M. Moreau[†], B. Shariat[†], and M. Beuve[†]

November 30, 2010

Abstract

We present a method to track a patient and the equipment in a radiotherapy treatment room, by exploiting the information in the treatment plan, enriched with other elements such as visual, geometric and "semantic" information. Using all these information items, and a generic model, a virtual environment of the scene is created, with maximum precision. The images resulting from video sequences with several cameras are also used to confront the filmed information on the scene and its numerical representation. The method is based on the features of the scene elements, and on a fuzzy formalism [10]. The feasibility of the method is being quantitatively evaluated in the absence of treatment, to be further exploited in a module for external control by video in real conditions.

*K*eywords: Artificial, augmented, and virtual realities, Computer vision, Video analysis, Tracking, Feature evaluation and selection, Interactive systems, Fuzzy set.

1 Introduction

Our study concerns the development of an external control tool adapted to radiotherapy, focussing on the real requirement, expressed by medical doctors and radio-physicists, to have a global view of the treatment system, in order to follow the patient and all the equipment in the treatment room, be them mobile (robotized or not), or immobile.

Automated surveillance of the treatment room as a whole would allow to minimize, if not eradicate:

- the physical errors (manipulations, collisions, involuntary movements)
- the logical errors (patient's identity, positioning relative to a unique referential, parameters)

*Manuscript submitted to IEEE "Transactions on Biomedical Engineering"; November 19, 2010.

[†]CIFRE PhD research collaboration with DOSIsoft S.A., Cachan, France E-mail: miguel.portela-sotelo@liris.cnrs.fr

[‡]LIRIS Lab (Laboratoire d'Informatique en Image et Systmes d'Information), Lyon, France.

- the security risks (presence of staff during irradiation, opened door, ...).

Assistance during the patient positioning phase would also be facilitated, helping to reduce the duration of each treatment session, hopefully making it more "bearable" for the patient, and potentially allowing more sessions in a given time interval.

1.1 External control in radiotherapy

Radiotherapy was one of the first medical specialties to implement quality assurance, although by focussing mainly on equipment performance, and leaving the human factor aside, at least until very recently. The new trend is to a set up a risk-based approach, together with the means, through the medical staff, of the follow-up and the control over pre-, per- and post-treatment [18, 20, 3]. The ASN[24], in charge of the radiotherapy facilities in France, is a testimony of the role of human and organizational factors in the happening of incidents, a priority during the inspections performed in the French cancer treatment units.

Previous research propose modules for external control in radiotherapy as a tool for assisting the precise and automatic positioning of the patient. Although these modules focus on the patient, they attest the importance of external control systems in radiotherapy [8, 21, 15].

It is now possible to find in standard treatment rooms systems allowing a precise location of the tumor (*e.g.*, AlignRT [23], Polaris [35]), or accessories in the irradiation equipment (*e.g.*, CyberKnife [27], Novalis [33]). However, these devices do not take into account all the elements in the room that are liable to contribute to the treatment (robotically or manually), or are just there (furniture). We regard these systems as complementary to our own module, potentially providing it with more information on the patient in the future.

1.2 Tracking individuals and objects

Detecting and recognizing persons and objects in a video sequence are still very dynamic research areas. Indeed, various parameters of the filmed scene, such as changes in lighting or very fast displacements, lead to ambiguities in the process, requiring further computations to compensate for potential imprecision or lack of robustness.

Techniques for tracking individuals and objects by video may be arranged into two classes, according to whether they use an underlying model or not. Since a large quantity of information is at our disposal on the patient and the treatment room[1], that could allow to reconstruct a numerical model of reality, we decided to concentrate on model-based techniques.

Such techniques are known to be more precise and robust than the others, since the model contributes to provide, at each moment, a great number of information elements, at all possible levels. Although their main disadvantage lies in the complexity of the computations, they are better-suited to our needs. Radiotherapy treatment rooms are highly controllable environments, allowing more reliable predictions of possible events, and hence, a reduction of the potentially large number of computations to be performed. Furthermore, the constraints

[1] some of which may be very fine, like for instance, those pertaining to the patient as retrieved from imaging performed before treatment.

and medical processes in radiotherapy treatment naturally ensure the availability of a series of factors to facilitate the computing effort, in particular through research space reduction during computer processing.

2 Video tracking as an efficient tool for external control in radiotherapy

We are interested in the secured supervising of the whole treatment room, in order to check the conformance of treatment plans as generated by the TPS (Treatment Planning System), and to reduce the number of potential hazards. Since the environment is theoretically well under control, the large majority of informational data is already present in the treatment plan. Exploiting these constraints helped reduce the run time performance of our complex tracking and supervision methods. We have good hopes that they may even allow to reach interactive time in the future.

2.1 The patient and the objects

During the pre-treatment phase, medical CT imaging is used to determine the size, the shape and the location of the tumor. Images are transferred over to the TPS, in which the radiotherapist (or the oncologist) identifies the tumor and the sound regions to be avoided by the beam. Next, a treatment plan is designed by the radio-physicist with the TPS, through the indication of the ballistics, the prescribed dose at the tumor while "sparing" as precisely as possible the sound tissues around using safety (margin) volumes deduced from the set of CT studies, the characteristics of the tumor and the beam. At the beginning of each treatment session, the patient is invited to lie on the couch in the same position as the one during the initial medical imaging session. Contention devices allow a better, but not 100%, security that the patient is correctly positioned and immobilized during all the sessions.

For the visual aspects of our module, the treatment plan provides information on the patient's "surface" in the zone to be irradiated, as well as the "entry point" (port) of the irradiation beams relatively to this surface. These elements, related to fixed information (description of the empty room, of the machines and equipment, of fixed spots or devices in the room, such as lasers, isocenter, etc.), allow us to automatically generate an augmented reality environment similar to the real treatment room at any time during the session. For this, we use the generic model that we have developed [16] that uses XML format descriptor files, so as to transmit information to the movement tracking module, and exploits files with other format files (DICOM[2], VRML[3], and others).

2.2 Main objective of the external control system

After the patient has been positioned, but before irradiation, it is possible to derive a configuration of the augmented reality environment equivalent to that of the treatment room (under the condition that the positioning procedure did

[2]Digital Imaging and COmmunications in Medicine
[3]Virtual Reality Modeling Language

not fail). From then on, we assume that neither the patient nor any of the objects inside the room will move on its own (outside programmed scenarios for potential robots, or beam arms), and that no person nor object will enter or leave the room, the door(s) of which will stay closed during the rest of the session until its termination. The exact goal of the surveillance of the treatment room is to check that this assumption is always satisfied, and if this is not the case, to inform the medical staff, so something is done to prevent any source of hazard.

We focus on the following:

1. the detection of unprogrammed displacements (relatively to the treatment plan instructions and a given threshold), either of the (head, shoulders, arms etc of the) patient, or the objects within the room,
2. the detection of unplanned persons or objects during the session,
3. the observance of small patient movements (like breathing cycles).

3 Our approach

We use a generic model that provides geometric and visual information to help detect and track several elements in the treatment room. This generic model uses 3D descriptors that are numerically defined by projection on the cameras' image planes, and correspond to 2D primitives (edges, feature points, patterns, textures, etc.).

The multi-primitive aspect of the method yields more precise and robust results during detection and tracking on video images [11]. Furthermore, we exploit the mechanical constraints of the elements, as defined in the scene, and in particular the free and collision spaces [14]. Finally, a fuzzy logic-based strategy [36][10, 1] allows to merge the information provided by all usable descriptors in order to define the position in 3D space of any element in the room.

The general idea of the method is to let two environments interact: RE, the Real Environment (*i.e.*, the scene as filmed by cameras) and VRE, the Virtual Reality Environment (containing the numerical model of the scene: humans and objects).

The generic model we have developed is based on CSG (Constructive Solid Geometry) trees, augmented with visual, physical and semantical information. A scene graph containing the elements of the filmed scene (humans and objects) is constructed using this model, to produce the VRE.

To this VRE we add a numerical representation of the acquisition system. The virtual representation of a "real" camera ($Rcam_i$), designed as $Vcam_i$, contains (intrinsic and extrinsic) parameters of its real homologue, which allows, in particular, to generate a virtual image $Vimg_{(i,t)}$, that is equivalent to a real image $Rimg_{(i,t)}$ delivered by one given camera i at any time t (*i.e.*, of cameras $Vcam_i$ and $Rcam_i$, respectively), t being related to the frame rate of all the synchronized cameras. After the segmentation of 2D primitives on $Rimg_{(i,t)}$ and $Vimg_{(i,t)}$, we compare the contents of both images.

If the position of one scene element in RE corresponds exactly to that of its numerical representation in VRE, then, for any camera, the projection of the object on the image plane in RE is equivalent to the projection of the numerical

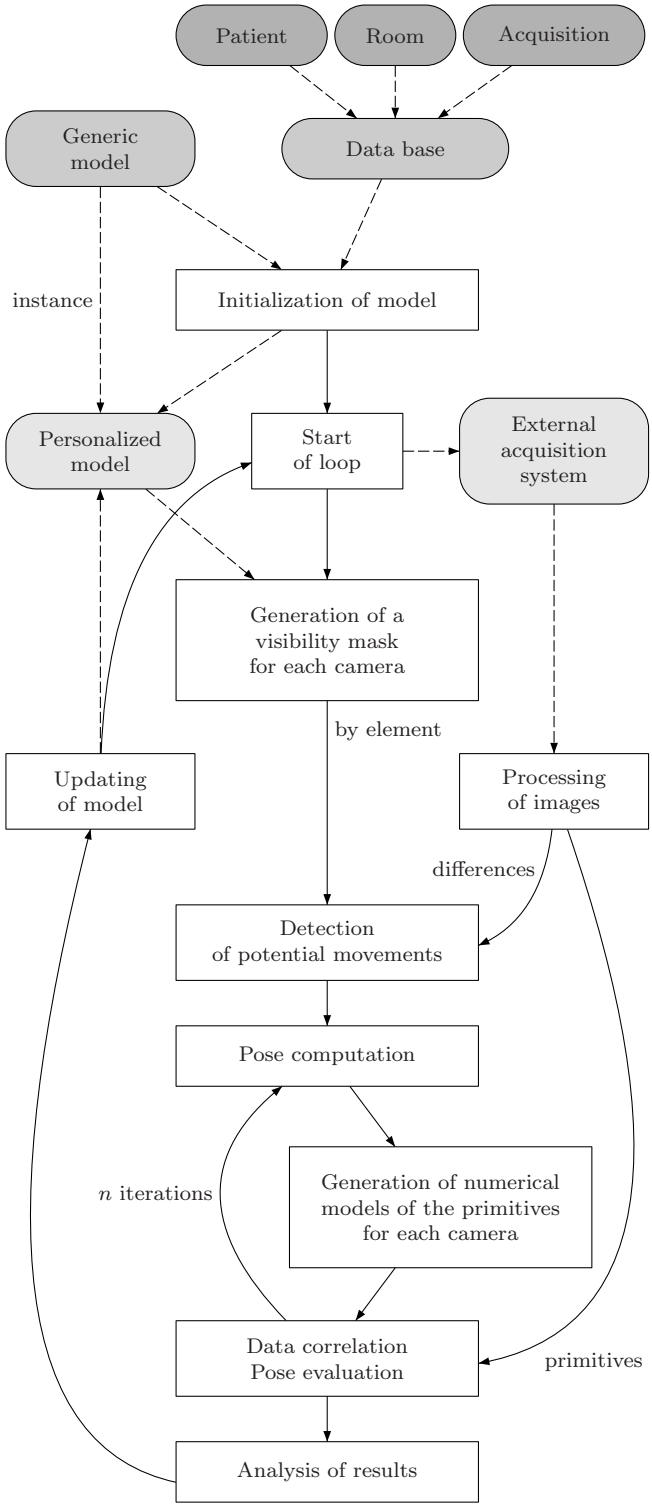


Figure 1: General flowchart for the method.

model of the object on the image plane of the numerical representation of the camera in VRE, obtained by simulation of the acquisition of an image of the scene by using the intrinsic and extrinsic parameters of the camera. This process is very close to the *Shape From Silhouette* (SFS) approach, but the other way round: in SFS, one detects the silhouettes (the sets of pixels containing the projection of the object on the image plane of a camera, seldom using the values of the pixels) of the object tracked on the set of cameras. The silhouettes are used to compute the object's bounding volume in space, which is then reconstructed relatively to this volume [12]. However, although this technique is more efficient in terms of complexity, it would not yield enough precision and robustness for our needs, since it does not take the object's topology into account.

This identity principle is frequent in model-based tracking methods using computer vision, which exploit the space of possible solutions for the objects' positions in space, in order to compute the finest possible approximation. Several research papers use such methods for tracking persons or parts of their bodies using video images and humanoid (volume or surface) models [19, 14, 5, 4], and also for tracking objects (with rigid or not models), frequently to apply them to "visual servoing"[4] [6, 9, 17]. Such research illustrate the feasibility of the technique in the realm of movement tracking, and their great advantage over the other published methods, as soon as one has an *a priori* knowledge of the target object. Since in our case the global variations of movements in the scene are limited and often very small, the position of a given element in it at time $t-1$ is frequently very close to its position at time t, and hence we may assume that we have, at every moment, an approximation of the position, to be improved by means of the features of the human or object.

The generic model with which we build the VRE is essentially composed of:

- element nodes,
- descriptor nodes that, associated to an element node, provide visual, physical and semantical information on the remarkable features of the element,
- operator nodes that, associated to several element nodes provide physical and semantical information related to the interaction between the associated nodes.

3.1 The 3D descriptors

To each element in the scene is attached a series of descriptors that will allow to retrieve the object in 3D space on the images. Hence, these descriptors are defined in correlation with the discriminating visual elements that may be derived from various segmentation processes (regions, frontiers, edges, corners, intersections between lines and curves, patterns).

We are not necessarily interested by the whole object, but by those specific features that are less difficult to find on videos, and that are such that the knowledge of a few of them on the image allows to position the object thanks to the underlying model.

[4] A technique using the information delivered by visual sensors, to perform position control for robots.

Descriptors represent referentials: they have their own coordinate system, and the constant matrix allowing to go from the coordinate system of one element and the descriptor is known in the model. The numerical representation of the descriptors also yields topological and colorimetric data that may be used for location purposes in the images.

As shown on the diagram of Fig. 1, the process loop is based on the generation of numerical models of the primitives for each camera and for each given pose, and each element. These models correspond to the projection of each descriptor on the image plane of each camera, in function of the intrinsic and extrinsic parameters.

This model is hence defined by an image $VSimg_{(i,t,j,k)}$ (on the image plane of $Vcam_i$ at time t for descriptor $Desc_{(j,k)}$ of element $Elem_j$), containing the projection of the descriptor. The set of non-null pixels in the image constitutes primitive $Vprim_{(i,t,j,k)}$. These models are saved for time $t+1$, if element j is observed to be static at time t.

$RSimg_{(i,t,j,k)}$ is computed by segmenting image $Rimg_{(i,t)}$ according to the type of the primitive representing $Desc_{(j,k)}$: for instance, for edge descriptors, we use a Canny-type filter; for the colorimetric descriptors, we use the back projection of the HSV-histogram to obtain a probability map, etc. As previously, the set of non-null image pixels after this operation constitute primitive $Rprim_{(i,t,j,k)}$.

Among all descriptors, the "smallest bounding volume" descriptor, that is associated to each element, plays an essential role when dealing with movement detection. Indeed, using the bounding volume of an element, it is possible to generate its silhouette on the image plane of each camera, and thus to define a zone of interest (at image level) where the object is known to have been at time $t-1$.

If one considers the silhouettes of all the elements and the depth value, with respect to the camera, of each pixel, for a given element, one gets a binary image $VSimg_{(i,t,j,k_v)}$, all of whose maximal values belong to the visible part of the element (denoted as "visibility mask" on Fig. 1). By computing the difference between $Rimg_{(i,t)}$ and $Rimg_{(i,t-1)}$, and by using $VSimg_{(i,t,j,k_v)}$ as a mask (at image level) of the pixels to consider, it is possible to detect a potential movement that other descriptors could afterwards help quantify.

3.2 The dissimilarity function

The model-based tracking methods may also be classified into two other categories: the stochastic ones, and the deterministic ones. Stochastic methods use estimation methods like Kalman filters, particle filters or other types of filters [22]. As already explained, the dynamics of the scene is quite low, making such methods less appropriate for our needs.

Deterministic methods are based on the iterative minimization of a cost function measuring the alignment of the model on real images. Two very well-known functions, based on objects frontiers, are the chamfer distance [2] and the Hausdorff distance [7]. As they are quite sensitive to aberrant (fluctuating) values, [14] suggested a function based on the measure of the area of non overlapping sections of surfaces for the global tracking of hand movement.

In order to apply this to multiple object tracking, we propose a "dissimilarity" function that combines the chamfer distance and surface overlap:

$$\rho(R,V) = \left(1 - \frac{|R \cap V|}{|R \cup V|}\right) \times \frac{1}{|CV|} \times \sum_{v_a \in CV} \underset{r_b \in CR}{min}\, d(v_a, r_b) \quad (1)$$

where $R = Rprim_{(i,t,j,k_d)}$, $V = Vprim_{(i,t,j,k_d)}$, $CV = CVprim_{(i,t,j,k_d)}$, $CR = CRprim_{(i,t,j,k_d)}$, and:

$$CVprim_{(i,t,j,k_d)} = Vprim_{(i,t,j,k_d)} \cap Vprim_{(i,t,j,k_c)} \quad (2)$$

where $Vprim_{(i,t,j,k_c)}$ is the edge primitive for the element. A similar formula exists for $CRprim$.

To explain the formula, let us consider two distinct poses ($3Dpose_a$, $3Dpose_b$), different from the target pose $3Dpose_t$. The function above allows to compute values for $3Dpose_a$ and $3Dpose_b$ in order to compare them with $3Dpose_t$ while searching for the optimum pose corresponding to the minimum value found in the process. The computing process works according to two mutually exclusive cases:

i when the silhouette of the element projected with respect to $3Dpose_a$ has no intersection with the target silhouette $3Dpose_t$ (refer to Fig. 2(a)), the first half of the above formula $\left(1 - \frac{|R \cap V|}{|R \cup V|}\right)$ corresponding to the notion of non-overlap, takes on the value 1, and the function behaves like a pure chamfer function:

$$\rho(R,V) = \frac{1}{|CV|} \times \sum_{v_a \in CV} \underset{r_b \in CR}{min}\, d(v_a, r_b)$$

which gives more weight to the translation component for convergence;

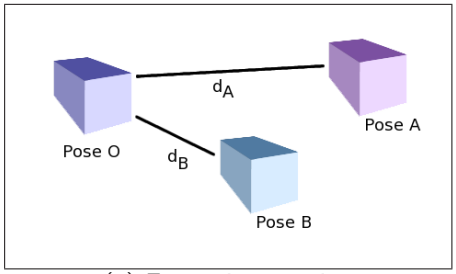
ii otherwise (Fig. 2(b)), the first half of the formula brings more precise information on the rotation component for convergence, due its relationship with surface.

Taking the silhouette of the primitive allows to overcome the problems related to occlusions and changes in lighting. Using the frontier of the object instead of only the frontier of the silhouette allows to ensure the robustness whenever the descriptor is only partially visible and when the silhouette is symmetrical relatively to one or more degrees of freedom of the object. For that reason, the silhouette and frontier descriptors are not included in the value assigned to a given pose.

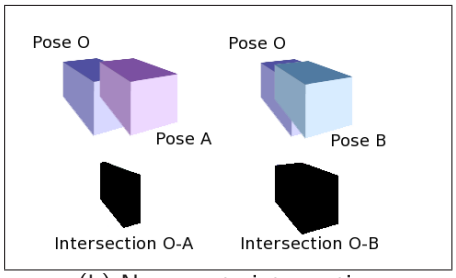
3.3 Merging the data issued by primitives

The next step in the chart of Fig. 1 is to compute the new object's pose. In general, most of the objects in the room are supposed not to move during irradiation. Among all the moving elements detected by our method at any time of the session, we are specifically interested in those that move differently from their programmed movements, or are neither programmed nor supposed to move.

A $3Dpose$ is defined by six parameters $\begin{pmatrix} x & y & z & \Psi & \theta & \varphi \end{pmatrix}$ corresponding to one translation and three rotations in 3D space (Euler angles). The starting



(a) Empty intersection



(b) Non-empty intersection

Figure 2: Illustration of the dissimilarity function.

point for the research is the position of the object at time $t-1$. The search space is centered around this point, and expanded according to the object's degrees of freedom, and the mechanical and spatial constraints added to the model. The errors imputed by scaling and shearing are compensated by using several cameras.

For a given pose in the search space, we generate the set of primitives $Vprim_{(i,t,j,k)}$. In order to alleviate the problems arising through occlusions and the noise due to the imprecision of the location method, we use fuzzy logic principles: fuzzy sets allow to merge heterogeneous data and take into account the imprecision generated by the acquisition and the processing of data.

We define two fuzzy sets (Fig. 3), one corresponding to the visibility at image level, and the other to the displacement in 3D space. These sets allow to associate a weight measuring the relevance of the pose approximation, and hence to compute the new position of element $Elem_j$.

The visibility is computed using the overlap of the silhouette and the visibility mask of $Vprim$. The weight $Pdesc_{(j,k,t)}$ gives a confidence rate in the pose for descriptor $Desc_{(j,k)}$, and results from the visibility of its primitives and the value given by the dissimilarity function 1:

$$Pdesc_{(j,k,t)} = \prod_{i}^{n} (Val^{v}_{(i,t,j,k)} \times \rho(R,V)) \tag{3}$$

where $Val^{v}_{(i,t,j,k)}$ is the value of the visibility of $Vprim_{(i,t,j,k)}$.

Let T^{3D} be the transformation from $3Dpose_{(j,t-1)}$ to $3Dpose_{(j,t)}$. The displacement is computed using the distance of T^{3D} relatively to the predefined threshold vector $S = \begin{pmatrix} x_s & y_s & z_s & \Psi_s & \theta_s & \varphi_s \end{pmatrix}$ (different for each element). T^{3D} is considered to be:

- below the threshold: if all its components are strictly inferior to their respective homologues in S,
- above the threshold: otherwise.

Two states are defined in the set corresponding to a displacement: staticity and effective movement, the values of which are given by an evaluation function between T^{3D} and S.

A pose $3Dpose_{(j,t)}$ for the element $Elem_j$ is evaluated from the $Desc_{(j,k)}$ descriptors whose weight $Pdesc_{(j,k,t)}$ is above a predefined threshold:

$$\sigma_{(t,j,k)} = \begin{cases} Pdesc_{(t,j,k)} & if\ Pdesc_{(t,j,k)} > S \\ 1 & else \end{cases} \quad (4)$$

Finally, the weight $P3Dpose_{(j,t)}$ for a $3Dpose_{(j,t)}$ is given by:

$$P3Dpose_{(j,t)} = (1 + max(Val_s, Val_m)) \times \prod_{k}^{n} \sigma_{(t,j,k)} \quad (5)$$

where Val_s and Val_m are, respectively, the values for the staticity, and effective movement states.

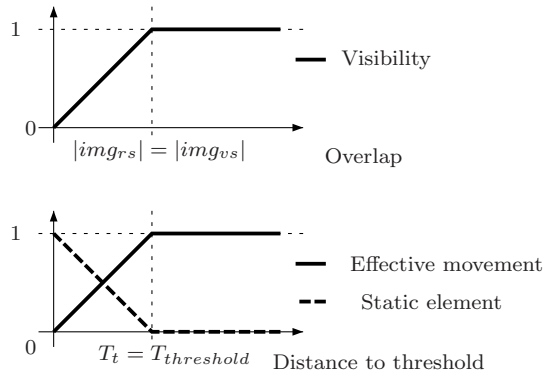


Figure 3: Associated fuzzy sets.

3.4 Computing the object's pose

Traversing the search space is done using the simplex approach proposed by Nelder and Mead [13]. This choice is justified by the simplicity of the implementation and, in particular, the fact that the derivative of the function to be minimized needs not to be known. Furthermore, this approach relies on the notion of simplex, a polytope of $N+1$ vertices in an N-dimensional space. This approach could hence be parallelized in order to reduce computing times, thanks to the fact that the space is searched in various directions.

Until now, we have explained how the characteristics pertaining to an element allow to retrieve its position in the scene. In order to accelerate the process of computing the pose of certain elements, it is possible to take advantage of the information from other elements of the scene. In order to do this, we have set

up a dual strategy based on priority queues in the model, and the $P3Dpose_{(j,t)}$ weight corresponding to pose $3Dpose_{(j,t)}$.

3.4.1 Using priority queues

A priority queue structure contains all the elements in the scene, that are progressively extracted in decreasing order, in function of:

- their importance: in the semantic sense, some elements should be tracked in priority (the best example being the patient during the radiotherapeutic treatment),
- their dynamics: because we have an *a priori* knowledge of the elements whose probability to move is highest (*e.g.* the robotized equipments, as opposed to the non-robotized ones),
- their free space: by taking into account the other elements with which a given element is liable to interact (for instance, the irradiation arm and its potential collision with anything around it).

We have organized the processing according to the flowchart on Fig. 4. The idea is to compute each element's pose according to its priority in the above sense. Taking into account the weight of the already evaluated positions allow to ponder their contributions to the computation for the pose of elements not yet dealt with. Similarly, if the information available is not sufficient to compute the pose of an element when extracted from the queue, it is set aside in a secondary prority queue where it stays until all the elements in the main one are processed, at which stage the two queues are exchanged and the process is resumed. This scheme is particularly efficient in gaining robustness in the case of partial occlusions. The evaluation at the "Sufficient information?" node in the chart on Fig. 4 is performed using two criteria:

1. the value of the $P3Dpose_{(j,t-1)}$ weight with regard to the value of a threshold that is decremented by a global, fixed predefined step, each time the two queues are swapped. More precisely, all the weights have values between 0 and some maximal value, and the step is a decimal fraction of this value. The smaller the threshold value, the less restrained the criteria: a zero value corresponds to no constraint. Hence no element may remain indefinitely in the queues.
2. the validity of the model's operators, that are totally dependent on the description given for the scene and of the interdependence of its elements. For instance, if it has been detected at a given iteration that the couch on which the patient is supposed to lie, has moved, but no patient movement has then been computed, then the logical operator linking couch and patient in the model is set from "disabled" to "enabled", and will be available to compute the patient's pose later in the process.

The constraints above guarantee, by construction reasons, that the double queue scheme will never enter an infinite loop. The overall algorithm for the dual queue scheme may now be summarized as follows (τ denotes the current threshold value):

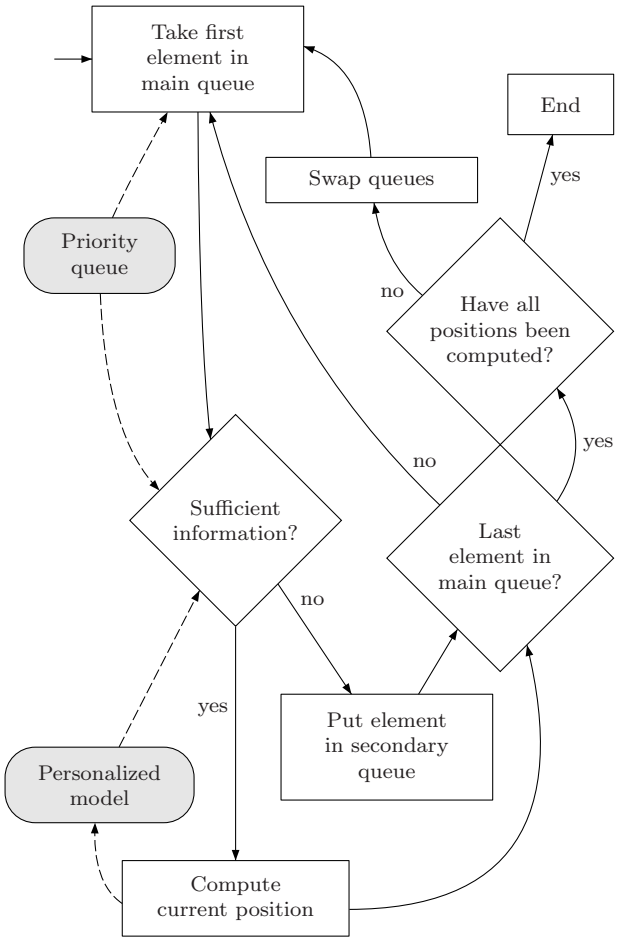


Figure 4: Using the main and secondary priority queues.

while at least one queue is not empty **do**
  while main queue is not empty **do**
    $e \leftarrow$ extremum from main queue
    update $weight(e)$
    if $weight(e) \geq \tau$ or e "depends" on other elements whose poses have all already been computed **then**
      compute $pose(e)$
    else
      add e to secondary priority queue
    end if
  end while
  swap queues
  decrement *tau* by fixed step
end while

4 Results

4.1 Evaluations

This method was implemented in C++ using `OpenCV`[34] libraries, `Coin3D`[26] et `dcmtk`[28].

We started with the evaluation on a sequence of computer generated images, which simulated a "real" acquisition, and was produced thanks to the generic model we have developed for the acquisition system. In particular, it allows to obtain a "goodness of fit" so as to deduce the system's precision independently of the processing of real images. The sequence was generated with a 640×480 resolution. The scene, a replica of one radiotherapy treatment room at the Léon Bérard Center[25] in Lyons, is composed of the irradiation arm, the treatment couch, the walls and floor.

During the simulation, for each object, and for each time interval (synchronized set of frames), a 3D pose is randomly chosen, the object is moved from its former pose to the new one, the displacement transformation is recorded, and an image is taken for each camera. The algorithm is then run on the resulting image sequences, in order to estimate the displacements of the filmed objects. Finally, we evaluate the differences between the original computed displacements and the displacements calculated from the image sequences by the algorithm, for each simulated position and each object. The results are displayed on the graphs of Figure 5.

These graphs show (on this particular case) that, in spite of minor fluctuations at a local level, the algorithm converges towards the global solution after a few iterations: error is not cumulative over time. The local behavior may be improved by adopting a windowing strategy taking the weights for computed poses into account.

4.2 Images

After the validating simulation phase described above, real image acquisitions were performed in the same treatment room. The acquisition system was com-

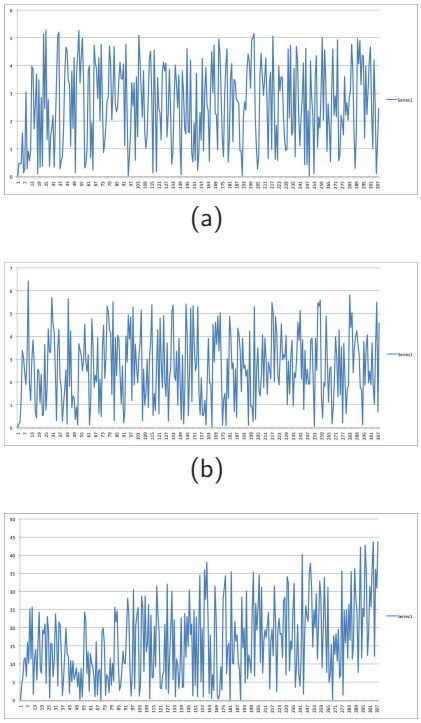


Figure 5: *Charts for the comparison between the generated poses and the ones computed from the filmed objects. (a): displacement differences for the irradiation arm (in degrees). (b): global displacement differences for the couch (in degrees); (c): displacement differences for the upper part of the couch (in millimeters).*

posed of two *Logitech QuickCam Sphere* webcams, yielding 640×480 resolution images at 20 frames per second. For obvious reasons and before we could benefit from clinical agreement, we processed the sequences in our laboratory after they were filmed *in situ*, although the final objective is to process them on the spot and during treatment.

The scene is manually initialized by placing the irradiation arm and the couch at a rest position (as defined for both devices). The model is configured to this very position. At time $t = 0$ there is an equivalence of positions that is evaluated and validated by the initialization phase of the method, before it starts learning the colors of every object and learning the "scene model" image (background) for the detection of movements that are to be fitted to a specific window for fixed time.

The irradiation arm is modeled as being made of two "region" descriptors and of a "characteristic point" descriptor. Each descriptor represents one section of the object. These parts, which are differentiated by their colors, were called, respectively: arm (region), source (region) and logo (characteristic point). The couch is modeled in two parts, this time in function of their degrees of freedom, called respectively: top (region) and middle (region).

In the images of Fig. 6 and 7, we illustrate the evaluation of a 3D pose on real images for the two objects considered. The two first images (a and b) of each figure are source images onto which the model is superimposed. Each column corresponds to the images from one camera. In each set of subsequent dual subimages:

- the left side image represents the distance map to the potential edges of the object section, essentially obtained through a Canny-type filter on the video image. The contours obtained by projecting the numerical model of the object section are superimposed on this image;
- the right side image contains the visibility mask (at the image level) of the object section, and the edges obtained by projecting the numerical model of the object section.

For the irradiation arm (Fig. 6), it is to be observed that the arm and source sections are accompanied by more noise, as the color of the object is close to that of the background. Hence, it is the logo section, more significant, that will help remove ambiguities. This shows the importance of the ponderation of each descriptor.

Observe that the "middle" section of the couch (Fig. 7) is not visible by the camera, on the right hand-side column (images b and f notably). On image f, the left subimage has all its values at a maximum. The computations represented by this image are hence not added to the 3D pose. This illustrates how occlusions are dealt with.

We have not managed real time yet. With two objects, the process yields 2 images per second. This could be improved by parallelising the search for the pose and by using graphic programming to generate the projection of descriptors.

5 Conclusion

We have set up tools allowing to create an augmented reality environment with the help of a generic model and descriptor files. Using the tracking method presented in this paper, we are ready to implement a control system in radiotherapy that will be capable of recognizing predefined scenarios to interact with the person operating the treatment room. Thanks to the scientific partnership established between DOSIsoft [29], ETOILE [30] and the Léon Bérard anti-cancer Center [25] in Lyons, we have already been able to lead the first experiments presented in these pages. Soon, we will have the opportunity to perform tests in real conditions (after validation of tests in equivalent, beam-less conditions). For this purpose, we are currently working on an extensive experimental protocol with the radio-physicists at CLB, and the project managers at DOSIsoft.

The progress in computer science, physics and robotics, in particular, leads to automating medical treatments. As a consequence, new equipments are incorporated every day in treatment rooms all over the world, and not only in radiotherapy. These equipments require knowing positions relative to the patient, as is the case for treatments in radiotherapy. Although such scenarios are today getting more complex, notably through the presence of medical staff during treatments, we are confident that systems like ours could easily extend to any clinical situation.

Acknowledgements

This PhD research is funded by a convention between DOSIsoft [29] and the LIRIS [32] Lab, in the setting of the INSPIRA [31] project, also implying the Lon Brard Center [25] in Lyons, and ETOILE [30] for the scientific supervision.

The authors wish to thank Ms. Chantal Ginestet and Pauline Dupuis, medical physicists at Lon Brard Center, for their availability and dedication, together with Baptiste Germain, Jean-Christophe Diaz, and M. Hanna Kafrouni at DOSIsoft, and the ETOILE project.

References

[1] Maysam F. Abbod, Diedrich G. von Keyserlingk, Derek A. Linkens, and Mahdi Mahfouf. Survey of utilisation of fuzzy technology in medicine and healthcare. *Fuzzy Sets and Systems*, 120(2):331 – 349, 2001.

[2] G. Brogefors. Hierarchical chamfer matching: A parametric edge matching algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10:849–865, 1988.

[3] P. Francois and E. Lartigau. Analyse des risques en radiothérapie - risk analysis in radiotherapy. *Cancer/Radiothérapie*, 13(6-7):574 – 580, 2009.

[4] D.A. Gomez Jauregui and P. Horain. Region-based vs. edge-based registration for 3d motion capture by real time monoscopic vision. In *MIRAGE09*, pages 344–355, 2009.

[5] Patrick Horain and Mayank Bomb. 3d model based gesture acquisition using a single camera. In *WACV '02: Proceedings of the Sixth IEEE Workshop on Applications of Computer Vision*, page 158, Washington, DC, USA, 2002. IEEE Computer Society.

[6] Seth Hutchinson, Greg Hager, and Peter Corke. A tutorial on visual servo control. *IEEE Transactions on Robotics and Automation*, 12:651–670, 1996.

[7] Daniel P. Huttenlocher, Gregory A. Klanderman, Gregory A. Kl, and William J. Rucklidge. Comparing images using the hausdorff distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15:850–863, 1993.

[8] L.S. Johnson, B.D. Milliken, S.W. Hadley, C.A. Pelizzari, D.J. Haraf, and G.T.Y. Chen. Initial clinical experience with a video-based patient positioning system. *International Journal of Radiation Oncology Biology Physics*, 45(1):205–213, 1999.

[9] A. Krupa, J. Gangloff, C. Doignon, M. F. de Mathelin, G. Morel, J. Leroy, L. Soler, and J. Marescaux. Autonomous 3-d positioning of surgical instruments in robotized laparoscopic surgery using visual servoing. *Robotics and Automation, IEEE Transactions on*, 19(5):842–853, 2003.

[10] M Mahfouf, M.F Abbod, and D.A Linkens. A survey of fuzzy logic monitoring and control utilisation in medicine. *Artificial Intelligence in Medicine*, 21(1-3):27 – 42, 2001. Fuzzy Theory in Medicine.

[11] E. Marchand and F. Chaumette. Virtual visual servoing: A framework for real-time augmented reality. In G. Drettakis and H.-P. Seidel, editors, *EUROGRAPHICS 2002 Conference Proceeding*, volume 21(3) of *Computer Graphics Forum*, pages 289–298, Saarebrücken, Germany, September 2002.

[12] Brice Michoud, Erwan Guillou, and Saida Bouakaz. Human model and pose Reconstruction from Multi-views. In *International Conference on Machine Intelligence (ACIDCA-ICMI)*, November 2005.

[13] J. A. Nelder and R. Mead. A simplex method for function minimization. *The Computer Journal*, 7(4):308–313, January 1965.

[14] Hocine Ouhaddi and Patrick Horain. 3d hand gesture tracking by model registration. In *Proceedings of International Workshop on Synthetic - Natural Hybrid Coding and Three Dimensional Imaging (IWSNHC3DI'99)*, pages 70–73, September 1999.

[15] S. Pinault, G. Morel, R. Ferrand, M. Auger, and C. Mabit. Using an external registration system for daily patient repositioning in protontherapy. IROS 07, IEEE/RSJ International Conference on Intelligent Robots and Systems, 2007.

[16] Miguel Portela Sotelo and Jean-Michel Moreau. Développement d'un module de suivi de mouvements et localisation d'un patient par rapport à un ensemble d'équipements dans une salle de traitement par radiothérapie. Technical Report RR-LIRIS-2009-035, LIRIS UMR 5205 CNRS/INSA de Lyon/Université Claude Bernard Lyon 1/Université Lumière Lyon 2/Ecole Centrale de Lyon, October 2009.

[17] Muriel Pressigout, Andrew I. Comport, Eric Marchand, and Francois Chaumette. Real-time markerless tracking for augmented reality: The virtual visual servoing framework. *IEEE Transactions on Visualization and Computer Graphics*, 12(4):615–628, 2006.

[18] J. C. Rosenwald. Sécurité en radiothérapie: le contrôle des logiciels et des systèmes informatiques - safety in radiation therapy: quality assurance of computerized system. *Cancer/Radiothérapie*, 6(Supplement 1):180 – 189, 2002.

[19] Cristian Sminchisescu and Bill Triggs. Covariance scaled sampling for monocular 3d body tracking. In *CVPR*, pages 447–454, 2001.

[20] D. Talandier, A.-T. Tajahmady, and S. Woynar. Sécurité en radiothérapie: résultats de trois ans d'expérience avec la mission nationale d'expertise et d'audits hospitalier (meah) - radiotherapy safety : Meah evaluation at three years. *Cancer/Radiothérapie*, 13(6-7):461 – 465, 2009.

[21] S. Tao, A. Wu, Y. Wu, Y. Chen, and J. Zhang. Patient set-up in radiotherapy with video-based positioning system. *Clinical Oncology*, 18(4):363–366, 2006.

[22] A. Yilmaz, O. Javed, and M. Shah. Object tracking: A survey. *Surveys*, 38(4 2006):13, 2006.

Web References

[23] AlignRT. Technology enhances the process of patient setup, surveillance and respiratory gating during radiation therapy.

[24] ASN: French National Nuclear Security Agency.

[25] Centre Léon Bérard. Private, non-profit, comprehensive hospital dedicated to cancer treatment and research.

[26] Coin3D. An OpenGL based, retained mode 3D graphics rendering library.

[27] CyberKnife. A frameless robotic radiosurgery system.

[28] DCMTK : DICOM ToolKit. A collection of libraries and applications implementing large parts the DICOM standard.

[29] DOSIsoft. French leader in treatment planning systems for radiotherapy.

[30] ETOILE: Espace de Traitement Oncologique par Ions Légers dans le cadre Européen.

[31] INSPIRA: INformatics for the Safety of Processes and Installations in RAdiotherapy.

[32] LIRIS: Laboratoire d'Informatique en Image et Systèmes d'Information.

[33] Novalis. Radiosurgery system.

[34] OpenCV: Open Source Computer Vision. A library of programming functions for real time computer vision.

[35] Polaris. An optical measurement system.

[36] Steve Kaelher. Fuzzy Logic Tutorial, 1993.

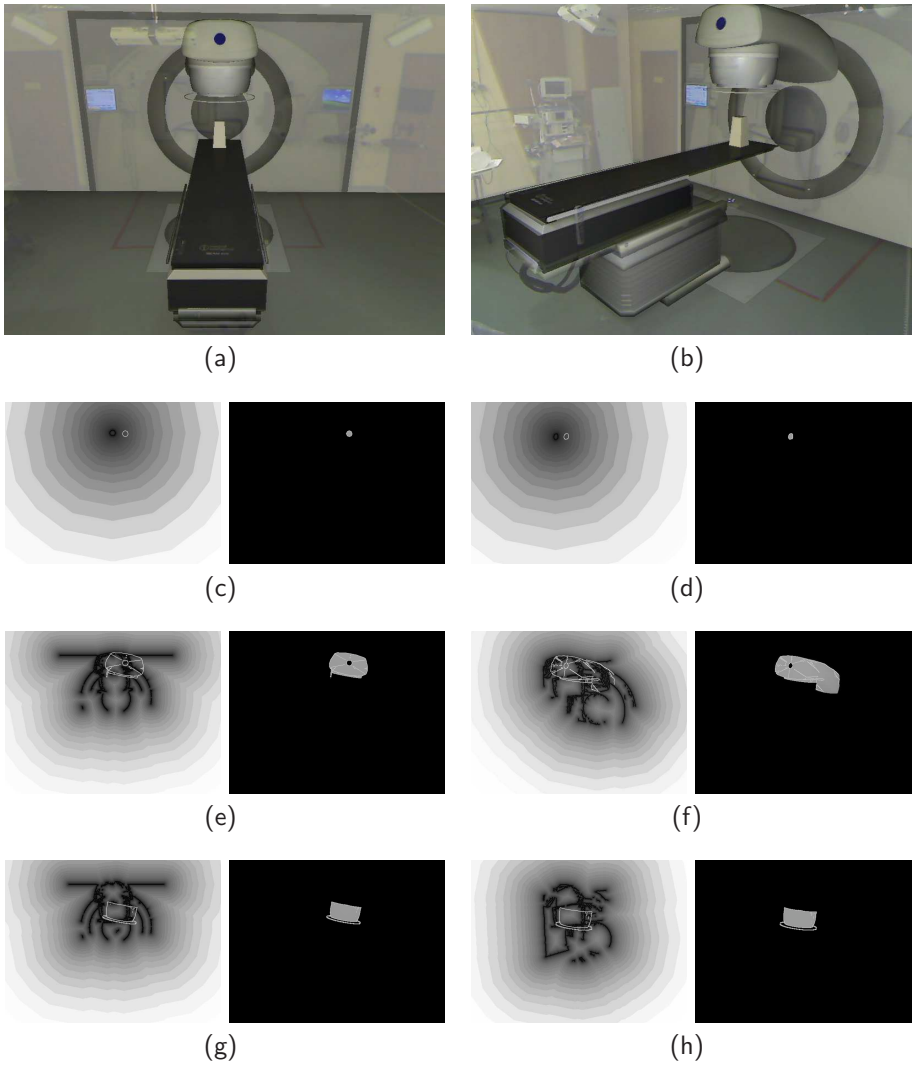


Figure 6: Images resulting from our module – including the two real images (at the top), the gantry logo, its arm and the beam source.

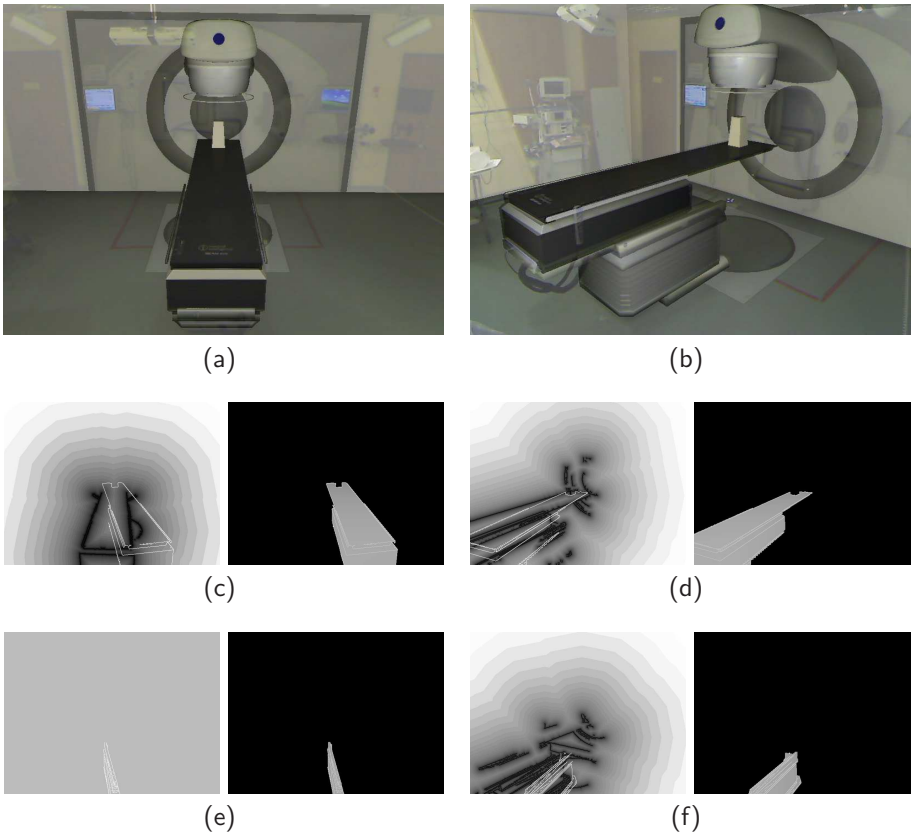


Figure 7: Images resulting from our module – bottom elements, including the two real images (at the top) and the couch.