# Region of Interest and Multiresolution for Volume Rendering

Sébastien Piccand, Rita Noumeir, *Member, IEEE*, and Eric Paquette, *Member, IEEE*

***Abstract*— Medical image interpretation is facing an important challenge resulting from the continuously increasing amount of imaging data. Innovations in medical image visualization are necessary to assist the radiologist in interacting and navigating effectively large multi-dimensional imaging sets.**

**We propose a novel wavelet splatting approach for multiresolution 3D visualization. Our method renders the context with a low resolution at first, and then subsequently refines it progressively to attain full resolution, while ensuring that a specific region of interest is rendered at full resolution at all times. It is based on the splatting approach for its computational efficiency and uses the localization property of the wavelet transform to simultaneously render a full resolution region of interest with a coarser context. Lighting calculations are used in the preprocessing stage to enhance the quality of the visualization. A special data structure that is based on a zero-tree model is used to manipulate the region of interest more easily. The speed-up achieved reaches a factor of 30 compared to the time needed to display the full resolution data. By achieving effective 3D rendering, we bring an element of solution to the problem of the image overload.**

***Index Terms*— 3D visualization, large data sets, medical imaging, multiresolution, navigation, region of interest, volume rendering, wavelet splatting**

## I. Introduction

ONE of the most important challenges facing medical image interpretation results from the continuously increasing amounts of imaging data. The increasing size of imaging data is due mainly to enhanced capabilities of imaging modalities such as multislice spiral-computed tomography. More precise data can be obtained with less time. Also, as scanning is becoming more effective, more data-intensive scanning protocols, such as whole body scanning, are becoming a trend. Therefore, the biggest challenge facing a radiologist is interpreting this huge amount of data precisely and effectively. According to the Society for Imaging Informatics in Medicine (SIIM) [1], "...image overload may be the single biggest challenge to effective, state-of-the-art practice in the delivery of consistent and well-planned radiological services in health care today." The challenge is to develop a completely new paradigm for looking at information and image data overload. Participants in the Transforming the Radiology Interpretation Process (TRIP) initiative [1] initiated by SIIM have identified innovations in medical image visualization as necessary to make progress in managing the ever-increasing volume of data. To illustrate the challenges, consider that a single scan from a modern computerized tomography (CT) scanner can generate over 2,000 images with an uncompressed memory size of more than 1 gigabyte, assuming 2,048 images with $512 \times 512$ pixels at 2 bytes/pixel. These images usually represent parallel planar cuts within a volume of interest. On a typical radiology workstation, images can be displayed in stack mode where navigation is possible either sequentially, by scrolling through the slices, or non-sequentially (and more effectively) using thumbnails or multi-planar visualization. Image analysis by a specialist requires a mental 3D reconstruction of the data, which is a very difficult and subjective task. More advanced visualization techniques, such as volume rendering [2]–[4] enable volume representation of the data to assist in the 3D volume reconstruction in an objective way, by displaying the data in a translucent manner. However, due to the large amount of data to be processed, these techniques require time-intensive computations that make efficient and interactive visualization impossible. Innovative visualization techniques are therefore needed to assist the radiologist: in approaching the growing amounts of information available to interpret, in interacting with large data sets, and in keeping pace with innovations that promise new horizons in medical imaging, such as effectively navigating five dimensions by blending multi-modal dynamic images as in cardiac PET-CT and functional cardiac MRI imaging [5]. While interpreting the imaging data, radiologists usually follow a model for visual search that postulates a pre-attentive global analysis followed by fixations and discovery scanning. This model was used by Kundel [6] to differentiate errors of search, recognition, and decision making. Therefore, in the focusing phase, only some structures in the data are of interest. These structures typically occupy a small percentage of the data, but their analysis requires contextual information like locations within a specific organ or adjacency to sensitive structures [7]. Therefore, while focusing on a particular region of the data, designated as a Region Of Interest (ROI), contextual information (called Context) surrounding that region is important. However, the same amount of detail is not required for the Context and the ROI. A multiresolution representation of the data enables the exploration of low resolution representation of the data while focusing on full resolution ROI, therefore achieving efficiency.

We propose a multiresolution visualization method that renders the Context with a low resolution at first, and subsequently automatically refines it progressively to attain full

Manuscript received July 15, 2006; revised June 27, 2007. This work was supported in part by Discovery Grants from the Natural Sciences and Engineering Council of Canada (NSERC).

S. Piccand and R. Noumeir are with LIVIA, Electrical Engineering Department, École de technologie supérieure, Montréal, Canada.

E. Paquette is with LESIA, Software and IT Engineering Department, École de technologie supérieure, Montréal, Canada.

resolution. Our approach achieves efficiency in 3D rendering by benefiting from the assumption presented in [6], [7] that presumes the user focuses on a small fraction of the data at a time, while navigating the whole dataset. During the interpretation process, the user interacts with the volume, by changing its scale or rotating it, for example, and therefore moving the region of interest. Even though we did not address the problem of determining an ROI, our method brings a contribution to the problem of interacting with a large dataset by proposing a novel rendering algorithm to achieve efficient 3D visualization. The new algorithm allows 3D rendering with a full resolution ROI. More work is needed to explore intelligent 3D ROI selection during the user interaction with the volume. Intelligent ROI selection is not new to the medical imaging field. Automatic two-dimensional (2D) ROI selection and a novel 2D rendering method have recently allowed efficient 2D medical image visualization [8], [9].

Simultaneous availability of detail and context at different resolution levels has been explored in [10] with a detail-in-context-technique for the purpose of navigating and viewing planar tomographic imaging data. Lamar [11] proposed a multiresolution approach to render the ROI at full resolution and the rest of the data—the Context—at a lower resolution. The multiresolution representation of the data was achieved using octrees and a pyramidal structure. On the other hand, wavelets have been proven to be more efficient than pyramidal representations for a multiresolution representation of data [12]. Wavelets have been very useful in image compression in general. They have also been used to provide multi compression levels for ROI and Context to achieve efficient image transfer and storage. Wavelets are the basis of the JPEG 2000 Interactive protocol (JPIP) [13] recently adopted by Digital Imaging and Communications in Medicine (DICOM) to enable an imaging server to transmit only the portions of a JPEG 2000 image that are applicable to the client's needs, in order to achieve improvements in bandwidth efficiency and speed when performing certain image viewing tasks in a client/server environment, while reducing the storage and processing requirements of the client. Therefore, JPIP uses a wavelet representation of the images to allow a viewer application to remotely extract a particular region of the image, or a high- or low-quality version of the image. It also can be used to progressively forward images of increasing fidelity. Wavelets have also been successfully used in practice to enable interactively transfer and visualization of image portions at different level of resolutions [8], [9].

In this paper, we are interested in multiresolution 3D rendering. In order to speed up the 3D rendering of large data sets, we propose a method that allows progressive refinement of Context rendering while ensuring that a specific ROI is rendered at full resolution at all times. After the Context is rendered with all possible refinements, the end result is a 3D representation of the data set at full resolution. By trading context resolution for efficiency, our method allows rapid rendering of an initial lower resolution Context with full resolution ROI. Fast rendering is very practical when the user is navigating or manipulating the volume, such as rotating it. Progressive refinement as proposed by [14] but applied only on the context allows a full resolution rendering of the complete data set to be achieved when the user is focusing. The wavelet transform localization property, along with its extension to represent volumetric data, make this technique suitable to our objectives. Wavelets have been used in the work of Krishnan et al. [15], where the data were divided into blocks and where each block was represented independently using the wavelet transform. The blocks were rendered at full or low resolution depending whether they belonged to the ROI or to the Context. While this method achieves multiresolution rendering, it presents a disadvantage, as the use of blocks requires reconstructing them before rendering. Moreover the reconstruction had to deal with edge effects, either by overlapping blocks or by using different wavelet filters for borders, which tends to be very time-consuming.

On the other hand, wavelet splatting [14], [16]–[19] has been proven to be a very efficient way to use wavelets for the multiresolution rendering of the data. It consists of representing the whole data with the wavelet transform, thus avoiding border artifacts as the data are usually inside an empty space. Rendering is then achieved directly in the wavelet domain and the reconstruction of the data is carried out during the rendering phase. The focus of wavelet splatting was on data compression. The notion of ROI, using the localization property of the wavelet transform, was introduced as a way to further compress the data in a preprocessing stage by filtering the wavelet coefficients, but not as an interactive tool to navigate inside the data.

In this paper, we use the wavelet splatting approach in a novel way. We propose a new method to use the localization property of the wavelet with the splatting approach in order to achieve effective multiresolution 3D volume rendering. As splatting enables effective 3D rendering and the localization characteristic of the wavelet transform enables multiresolution schemes, combining both in a novel method allows us to provide a new 3D rendering method that offers both effectiveness and multiresolution at the same time. Our new method brings an element of solution to the image overload problem facing medical imaging specialists. It assists in effectively interacting and navigating 3D imaging volumes by rendering large medical data sets rapidly with a region of interest that can be interactively moved.

This paper extends our previous work [20] as we present here an efficient way to handle the ROI and the data located in front of or behind the ROI when the data is stored as trees of wavelet coefficients. This paper also discusses the optimal level of decomposition to achieve the best speed-up while keeping enough information in the context to be able to localize the ROI. We show that, using only a software method, the rendering speed already allows smooth interaction with the data.

## II. Multiresolution representation

A multiresolution representation of data is a structure representing the data at different levels of resolution. It is therefore possible to have access to each level of resolution. This is useful, for example, for segmentation where too high a resolution can lead to too many segmented objects. In the context

of volume rendering, the multiresolution representation of the data is used to accelerate the speed of the rendering during the manipulation phase by using low-level resolution. Once the view point is fixed, the high-level resolution is needed to achieve good analysis of the data.

This multiresolution representation is achieved using a Gaussian or Laplacian pyramid [11] or a wavelet decomposition of the data [15], [16], [18], [21], [22]. Wavelet decomposition leads to a more compact representation of the data. This is important because recent data acquisition techniques give data of very high precision, requiring a large amount of memory.

### A. Using wavelets

Using the wavelet transform, a data signal can be decomposed into two subspaces. One, $V_l$, is low resolution, whereas the other,$W_l$, consists of the details the low resolution lacks to be able to gain access to the high resolution. In the context of volume rendering, the signal is a 3D signal. A multiresolution representation can be obtained using a separable orthonormal wavelet basis and the tensor product.

For each subspace, a basis is created from the two basis functions $\phi$, the scale function, and $\psi$, the wavelet function, as in (1).

$$\begin{aligned}
&f_{0,0,\overrightarrow{0}}(x_1,x_2,x_3) = \phi(x_1)\phi(x_2)\phi(x_3),\\
&f_{0,1,\overrightarrow{0}}(x_1,x_2,x_3) = \phi(x_1)\phi(x_2)\psi(x_3),\\
&f_{0,2,\overrightarrow{0}}(x_1,x_2,x_3) = \phi(x_1)\psi(x_2)\phi(x_3),\\
&f_{0,3,\overrightarrow{0}}(x_1,x_2,x_3) = \phi(x_1)\psi(x_2)\psi(x_3),\\
&f_{0,4,\overrightarrow{0}}(x_1,x_2,x_3) = \psi(x_1)\phi(x_2)\phi(x_3),\\
&f_{0,5,\overrightarrow{0}}(x_1,x_2,x_3) = \psi(x_1)\phi(x_2)\psi(x_3),\\
&f_{0,6,\overrightarrow{0}}(x_1,x_2,x_3) = \psi(x_1)\psi(x_2)\phi(x_3),\\
&f_{0,7,\overrightarrow{0}}(x_1,x_2,x_3) = \psi(x_1)\psi(x_2)\psi(x_3),\\
&f_{l,i,(j_1,j_2,j_3)}(x_1,x_2,x_3) =\\
&\quad 2^{-\frac{3l}{2}}\, f_{0,i,\overrightarrow{0}}(2^{-l}x_1-j_1, 2^{-l}x_2-j_2, 2^{-l}x_3-j_3).
\end{aligned} \tag{1}$$

Once a 3D signal $I_v(\vec{x})$ has been decomposed, it can be reconstructed using (2).

$$\begin{aligned}
I_v(\overrightarrow{x}) = &\sum_{\overrightarrow{j}\in V_{M,0}} c_{\overrightarrow{j}} f_{M,0,\overrightarrow{j}}(\overrightarrow{x}) +\\
&\sum_{m=1}^{M}\sum_{i=1}^{7}\sum_{\overrightarrow{j}\in V_{m,i}} d_{m,i,\overrightarrow{j}} f_{m,i,\overrightarrow{j}}(\overrightarrow{x}).
\end{aligned} \tag{2}$$

where $\overrightarrow{x}$ is the position inside the 3D signal, $I_v$ the computed signal, $M$ the level of decomposition of the volume, $\left\{V_{M,0}, (V_{m,i})_{i\in[1..7]}\right\}$ the set of subspaces and $\left\{(f_{M,0,\overrightarrow{j}})_{\overrightarrow{j}\in V_{M,0}}, (f_{m,i,\overrightarrow{j}})_{m\in[1..M],i\in[1..7],\overrightarrow{j}\in V_{m,i}}\right\}$ the set of the basis functions of each subspace; and $\left\{(c_{M,0,\overrightarrow{j}})_{\overrightarrow{j}\in V_{M,0}}, (d_{m,i,\overrightarrow{j}})_{m\in[1..M],i\in[1..7],\overrightarrow{j}\in V_{m,i}}\right\}$ are the coefficients obtained through dot product between the original data and the basis functions.

## III. Region of Interest

In general, the physician focuses on an ROI when interpreting medical images. The structure of interest—a tumor or an organ, for example—often occupies less than 1% of all the data. Limiting the visualization to an ROI reduces the amount of data to process, but sacrifices contextual information (Context) which is useful for understanding the location of the ROI. Approaches that combine full-resolution ROI with a low-resolution Context promise to preserve the diagnostic value of the data while reducing the computation time during the navigation phase. This phase can not be achieved at full resolution because it requires extensive manipulation of the data by the user, and therefore would be too slow at full resolution. However, the user still has access to the full resolution at anytime during navigation by changing the rendering method, but at the cost of losing the fast interaction.

### A. Data structure for easy multiresolution access

First, the ROI must be defined. The simplest way is to consider a rectangular parallelepiped within the data space, so that its main axes are the axes of the data. To find which coefficients of the decomposed data are important to the reconstruction of the ROI, the localization property of the wavelet decomposition is used.

For example, when the 3D data are decomposed twice with the Haar wavelet, the coefficients needed to reconstruct an ROI of size $4 \times 4 \times 4$ voxels (i.e., 3D pixels) are: one low resolution coefficient $c_{2,0}$, seven detail coefficients of level 2 $\{d_{2,i,0}\}_{i\in[1..7]}$, and $7 \times 8$ detail coefficients of level 1 $\{d_{1,i,j}\}_{i\in[1..7],j\in[0..7]}$. This relation between the coefficients of each subspace can be used to represent the transformed 3D data with trees. The roots of these trees are the coefficients of the subspace of lower resolution. To each of these coefficients are linked the seven detail coefficients of the same level of decomposition. To each of the latter coefficients are linked the eight detail coefficients of lower level decomposition. This operation is repeated until all the coefficients have been added to the trees.

When the data is decomposed twice, the structure of Fig. 1 is obtained.

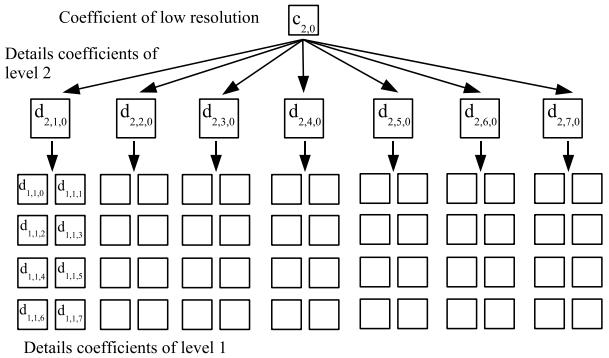



Fig. 1. Structure of the data when using a hierarchical representation of the coefficients

As all the trees have the same depth, this structure can be stored linearly as in Fig. 2, while maintaining easy access to the position of the roots.

This structure can also be used to easily compress the data by pruning the trees when the subtrees are only zeros. This

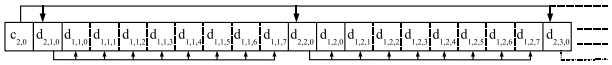


Fig. 2. Storage of the data

may happen when there is an empty or homogeneous space inside the data. In such a case, a particular value ZTREE is used to indicate that the subtree is empty. If this technique is used and if the data is stored as before, it is more difficult to retrieve the roots of each tree, without going through all the subtrees. In the case of rendering with an ROI, it is very important to have fast access to these roots as they give the position of the ROI. Therefore, we use a look-up table to store the position of the roots inside the data.

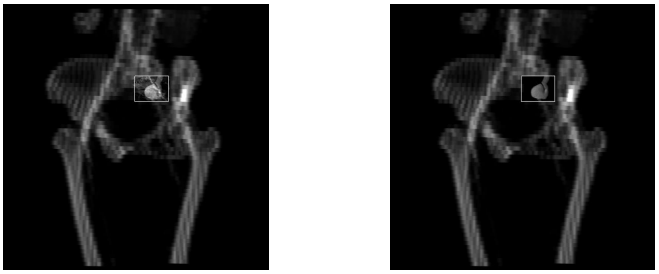
The structure using the hierarchical representation of the data is utilized in some compression techniques, such as Embedded Zerotree of Wavelets (EZW) by Shapiro [23]. As each tree enables the reconstruction of a different part of the data, we propose to add a look-up table for fast access to the roots of the trees, and thus to a particular region of the data.

### *B. Selection of the region of interest*

The ROI is selected by defining the position of the centre, and the size along the three axes of the volume. A border is displayed on the data to show where the ROI is. For the sake of simplicity, and because the user interface is not the goal of this paper, we did not consider the use of orthogonal views, which should help the user select the ROI.

### *C. Displaying the region of interest*

Once the ROI has been selected in the data space, one must be careful with the data that lies in front of and behind the ROI. The use of blending is a solution proposed by Lamar [11]. This is not suitable with the use of wavelet splatting with the Haar wavelet, as the Haar wavelet is sharp and gives blocky rendering at low resolution. This artifact is not a problem outside the ROI, as long as it gives us enough information on the localization of the ROI. We will discuss here two other possibilities: rendering all the data in front and behind the ROI at full resolution (Fig. 3(a)), or not rendering the data at all (Fig. 3(b)).



(a) Region of interest with data in front of and behind at full resolution

(b) Region of interest only

Fig. 3. Comparison of the rendering while displaying or not displaying the data in front of and behind the region of interest

In order to be able to hide the data that lies in front of and behind the ROI, the depth position of the data in the visualization space is required. For this purpose, we compute the "modified ROI," the rectangular parallelepiped whose main axes are those of the visualization space, and which encloses the "original ROI" (see Fig. 4).

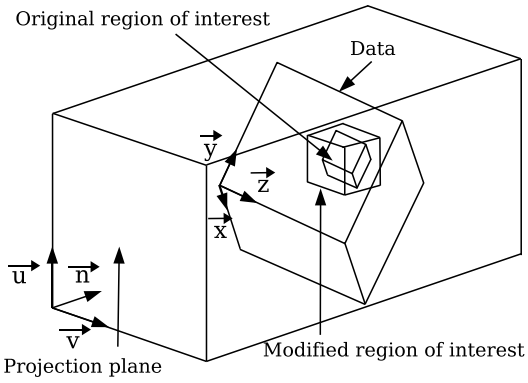



Fig. 4. Transformation of the region of interest from data space to visualization space. The coordinate system for the visualization space is $(\vec{u}, \vec{v}, \vec{n})$, whereas the one for the data is $(\vec{x}, \vec{y}, \vec{z})$

## IV. Implementation with wavelet splatting

The rendering process was implemented using the wavelet splatting method [19].

### *A. Rendering from wavelet space*

The wavelet splatting technique proposed by Gross and Lippert [16]–[19] comes from the fact that when we compute the values without taking opacity into account, it is possible to permute the projection and the reconstruction of a wavelet-decomposed volume. The reconstruction formula is presented in (3).

$$I_v(\overrightarrow{x}) = \sum_{\overrightarrow{j} \in V_{M,0}} c_{M,\overrightarrow{j}} f_{M,0,\overrightarrow{j}}(\overrightarrow{x}) + \sum_{m=1}^{M} \sum_{i=1}^{7} \sum_{\overrightarrow{j} \in V_{m,i}} d_{m,i,\overrightarrow{j}} f_{m,i,\overrightarrow{j}}(\overrightarrow{x}). \quad (3)$$

The way to project a volume onto a screen is presented in (4).

$$I(\overrightarrow{w}, \overrightarrow{n}) = \int_{s=S_{in}}^{S_{out}} I_v(\overrightarrow{w} + s\overrightarrow{n}) ds. \quad (4)$$

where $\overrightarrow{w}$ is the position on the screen, $\overrightarrow{n}$ is the direction of projection and $I(\overrightarrow{w}, \overrightarrow{x})$ is the corresponding intensity.

Combining (3) and (4), leads to (5).

$$I(\overrightarrow{w}, \overrightarrow{n}) = \int_{s=S_{in}}^{S_{out}} \sum_{\overrightarrow{j} \in V_{M,0}} c_{M,\overrightarrow{j}} f_{M,0,\overrightarrow{j}}(\overrightarrow{w} + s\overrightarrow{n}) + \sum_{m=1}^{M} \sum_{i=1}^{7} \sum_{\overrightarrow{j} \in V_{m,i}} d_{m,i,\overrightarrow{j}} f_{m,i,\overrightarrow{j}}(\overrightarrow{w} + s\overrightarrow{n}) ds. \quad (5)$$

As the coefficients do not depend on the direction of integration, the integration and reconstruction can be permuted, which leads to (6)

$$I(\overrightarrow{w},\overrightarrow{n}) = \sum_{\overrightarrow{j}\in V_{M,0}} c_{M,\overrightarrow{j}} \int_{s=S_{in}}^{S_{out}} f_{M,0,\overrightarrow{j}}(\overrightarrow{w}+s\overrightarrow{n})ds + \sum_{m=1}^{M}\sum_{i=1}^{7}\sum_{\overrightarrow{j}\in V_{m,i}} d_{m,i,\overrightarrow{j}} \int_{s=S_{in}}^{S_{out}} f_{m,i,\overrightarrow{j}}(\overrightarrow{w}+s\overrightarrow{n})ds. \quad (6)$$

Thus, the rendering process consists of a weighted sum of integrals that depend only on the wavelet basis chosen. These integrals are called wavelet footprints, and the rendering is done through the weighted accumulations of the footprints. Because of the scale property of the wavelet functions, only height footprints are generated whereas the others are derived from them using shifting and subsampling.

*1) Advantages of the rendering method:* This rendering method presents two main advantages. First, the rendering is processed directly from wavelet space. Therefore, no reconstruction, even local, of the data is necessary, leading to a low-cost processing task.

Moreover, this method allows the level of detail on the final images to be easily increased, by adding the footprints that correspond to the details of the wavelet decomposition. There is, therefore, no need to re-compute what has been done for the low resolution.

*2) Choice of the wavelet:* This method can be applied to many wavelet bases. Requiring that the rendering phase be as fast as possible implies that the footprints must be small. Indeed, the rendering phase can be decomposed into two main operations: the computation of the footprints, i.e., the integrals in (6); and the projection of the footprints, i.e., the sums in (6). The second operation is the most time-consuming. It depends linearly on the size of the footprints, which depends quadratically on the length of the support of the wavelet functions.

Moreover, in the context of rendering with an ROI, the number of coefficients in this region must be as small as possible for the rendering process to be fast, as the region is rendered at full resolution and requires many footprints to be splatted. Therefore, the size of the support of the wavelet must be small. The ROI is rendered at full resolution, so a high-order wavelet is not necessary to achieve good quality; indeed, it would increase only the quality of the Context.

Hence, the Haar wavelet is the better candidate as it is the wavelet basis with the smallest support size. According to Gargantini [24], it is also the best wavelet for perfect reconstruction.

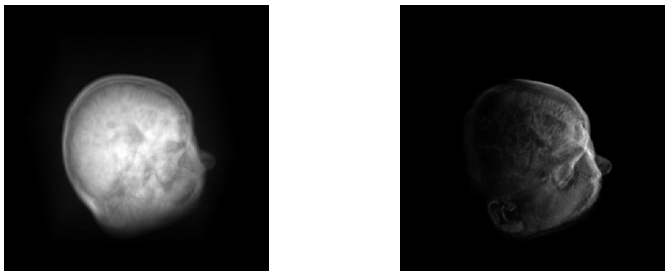
*3) Footprints generation and projection:* The footprints are generated using a splatting approach. As the Haar wavelet, which contains sharp discontinuities, is used, the splatting approach was preferred to a Fourier transform slice approach [14], [18]. Indeed, Horbelt [25] shows that the Fourier transform slice is better suited for wavelets of higher order.

First, the position of the footprints in the visualization space is computed. This is important because it enables us to know if the footprint is in front of or behind the ROI. Then, the projection is made using a bi-linear interpolation as the pixels of the footprint are not aligned with those of the screen pixels.

### B. Supports preprocessing

The wavelet splatting technique produces an efficient x-ray-like projection. To get an improved shading model, we propose using a simple model like the one from Phong-Blinn [26], [27], and with pre-integrated opacity like the one from Levoy [28] (Fig. 5).

Many other preprocessing tasks, such as blurring, edge detection, and histogram stretching, could be used. While improving the quality of the rendered data, they will not penalize speed during the manipulation phase. Indeed, whatever the complexity of the preprocessing tasks, the rendering phase stays the same.



(a) X-ray-like rendering (b) Enabling shadows

Fig. 5. Increasing quality of rendering with preprocessed lighting and shadows

## V. Results and discussion

The technique has been applied to a data set consisting of 1,200 slices of $274 \times 434$ pixels encoded as 16-bit integers. During the acquisition, the aorta was emphasized in order to show an aneurysm. This aneurysm represented a region of $64 \times 40 \times 40$ voxels or about only 0.07% of the whole data. After having been preprocessed and transformed into wavelet space, the voxels were then encoded as 32-bit floats to avoid, as far as possible, any effect due to compression. Compression is not the main purpose of this paper and is part of our future work.

Next, we will independently discuss the impact of the level of decomposition and the size of the ROI on the rendering speed.

### A. Influence of the level of decomposition

When increasing the level of decomposition, one can expect a speed-up of the rendering time. Indeed, while the footprints become larger by a factor of 4, the number of coarse coefficients decreases by a factor of 8. If only the footprint accumulation is taken into account, a speed-up by a factor of 2 can be expected.

But two other factors may influence the rendering speed: the percentage of zeros in the coarse coefficients may vary, and the computation of the footprints cannot be negligible if the level of decomposition is high (because the footprints will

become large). The impact of those problems is discussed in detail in the following paragraphs.

In the context of ROI-based rendering, an increased level of decomposition leads to an increased number of footprints to be displayed in the ROI to reconstruct the full resolution ROI. Therefore, if two levels of decomposition give similar speed-ups at low resolution, the coarser one should be preferred.

When the level of decomposition increases, the representation into trees may be more efficient and enable larger subtrees (corresponding to larger holes inside the data) to be eliminated. Thus, the compression ratio should also increase.

TABLE I

INFLUENCE OF THE LEVEL OF DECOMPOSITION ON SPEED-UP AND COMPRESSION RATIO. THE LEVEL 1 OF DECOMPOSITION IS TAKEN AS THE REFERENCE. THE RELATIVE NUMBER OF SPLATS USED AND THE PROPORTIONAL TIME TAKEN TO GENERATE THE FOOTPRINTS ARE ALSO GIVEN.

| Level of decomposition | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| Speed-up at low resolution | 1 | 3.7 | 7.2 | 7.5 | 3.8 |
| Compression ratio | 1 | 2 | 2.5 | 2.5 | 2.5 |
| Relative number of splats | 1 | $\frac{1}{5.2}$ | $\frac{1}{25.3}$ | $\frac{1}{113.9}$ | $\frac{1}{535}$ |
| Footprints generation time | 0.0054 | 0.124 | 1.696 | 13.89 | 57.14 |

Table I summarizes the relative speed-up and compression ratio when the level of decomposition is increased. The speed-up is computed against the level 1 of decomposition of the data, as only the decomposition level is of interest in this analysis. This shows that a level 3 of decomposition is the best choice for our implementation. After three levels of decomposition, as the compression ratio stays the same, no additional subtrees are eliminated, which shows that the data set does not have holes (or other homogeneous regions) of $16 \times 16 \times 16$ voxels. More important is the fact that the rendering process is not faster, and can even be slower. This comes from the two factors cited previously: the size of the footprints is too large not to take into account the footprint generation phase, as it takes more than 57% of the rendering time at level 5, and the number of non-zero coarse coefficients decreases by a factor of less than 8 (about 5).

For the rendering to be efficient, the ROI must be easily localized. Therefore, the quality of the image in the Context must be good enough to keep the information on the localization of the ROI. Fig. 6 shows the quality of the image at low resolution for levels 1, 2, 3, and 4 of decomposition of the data. Levels 1 and 2 provide good image quality but we saw that they were not the most efficient. Whereas a level 4 of decomposition leads to too coarse a representation, a level 3 of decomposition leads to an image quality still good enough to understand the data. It is possible, for example, to see the hip bones, and find which one is the left.

In the context of ROI-based rendering, an increased level of decomposition leads to an increased number of footprints to be displayed in the ROI. The speed-up from level 2 to 3 is important enough not to take into account the increased number of footprints of the ROI. The appropriate level with respect to the quality of the Context depends on the resolution of the input data, while the appropriate level with respect to speed depends on both the size of the ROI relative to the whole volume and the level of decomposition.

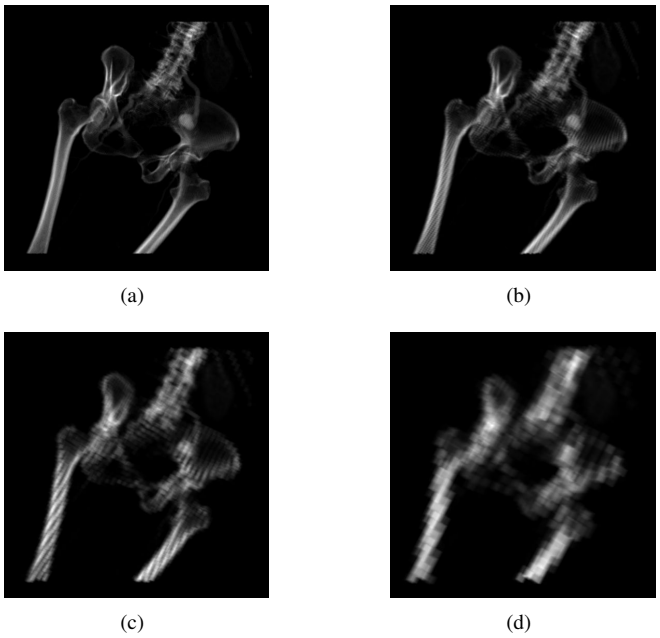


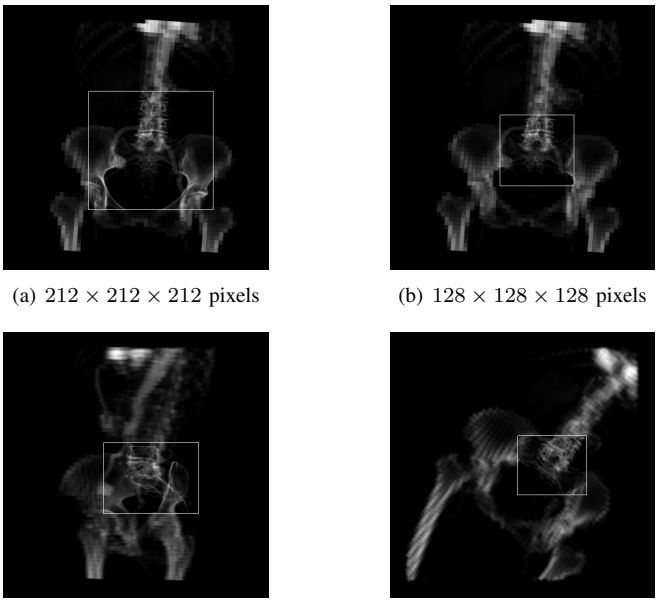
Fig. 6. Influence of the level of decomposition on the quality of image at low resolution from level 1 (a) to 4 (d)

### *B. Influence of the size of the region of interest*

The method proposed took as a hypothesis that the ROI was only a small part of the full data. The smaller the ROI, the faster the method is. An investigation was made to find out how small the ROI must be in order to achieve interactive manipulation of the data. An order of decomposition of level 3 was chosen, according to the results of the previous investigation. The ROI was chosen as cubic, centered on the spine and pelvis (because of the absence of big holes in this region), and was grown from $8 \times 8 \times 8$ voxels until it completely covered the data. At each step, the ROI was increased by adding 8 pixels to every dimension. The data in front of and behind the ROI was not displayed. The rendering speed was computed at multiple angles, and the one used to compute the speed-up was the median value (see Fig. 7). The speed-up is computed relatively to the time required to render the data with the wavelet splatting method, with a level 3 of decomposition, at full resolution (i.e., all the footprints have been projected). The results are presented in Fig. 8.

The curve is asymptotic as the ROI is growing. As the size of the ROI becomes larger than $15\%$, this corresponds to an ROI of more than $50\% \times 50\% \times 50\%$, which is so large that it would make more sense to render the whole data set at full resolution.

An ROI representing 0.8% of the whole data can already achieve good speed-up, as a factor of 10 is obtained in this case. A value of 0.8% may sound small, but the ROI has considerable size. It represents a parallelepiped that is $20\% \times 20\% \times 20\%$ of the whole dataset, that is $240 \times 55 \times 87$ voxels. For the aneurysm ROI that represents less than 0.1%



(a) $212 \times 212 \times 212$ pixels (b) $128 \times 128 \times 128$ pixels

(c) $128 \times 128 \times 128$ pixels (d) $72 \times 72 \times 72$ pixels

Fig. 7. ROI used to compute the speed-up at different sizes and angles

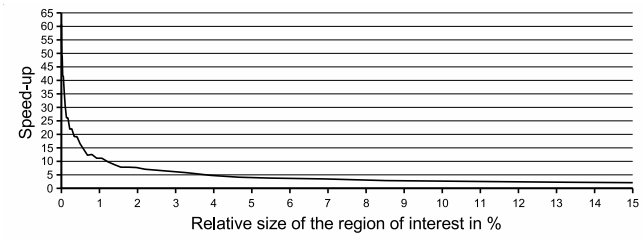



Fig. 8. Influence of the size of the region of interest on speed-up

of the data, a region of $64 \times 48 \times 48$ voxels, a speed-up factor of 30 is obtained. In our implementation the full data was rendered at a speed of 0.3 images per second on a 1.6 GHz Pentium-M. Using the region of interest around the aneurysm led to a speed of 9 images per second which is already fast enough to move the data interactively.

For comparison, with the normal splatting method without opacity integration during the rendering phase (to make the rendering faster with an image quality equal to the one used in wavelet splatting), the data were rendered at 0.16 images per second on the same computer, i.e., about two times slower than the full resolution rendering with wavelet splatting. It is also worth noting that without using a specific data structure for the normal splatting method, the memory limit of the computer (512MB) required to use a quantization of the data, which leads to a little loss of information.

## VI. Conclusion

We are proposing a novel volume rendering method that uses a multiresolution approach for rendering a full resolution Region of Interest within a low resolution Context while allowing the context resolution to improve progressively towards full resolution. Our method achieves fast 3D rendering of large data sets by allowing an initial Context to be rendered with a low resolution while ensuring that a specific ROI is rendered at full resolution at all times. Subsequently, Context is refined to achieve a complete full 3D rendering. Our method brings an element of a solution to the significant challenge facing medical image interpretation as a result of the constantly increasing amounts of imaging data. It is based on the idea that the radiologist often focuses on an ROI when interpreting medical images. By limiting the visualization to an ROI without sacrificing the contextual information, our method reduces the computational time and achieves efficiency in volume rendering, while preserving the diagnostic value of the presentation. The problem of determining an ROI has not been addressed, and additional work is needed to explore intelligent 3D ROI selection during the user interaction with the volume.

The proposed method uses a 3D wavelet decomposition of the volumetric data. The wavelet coefficients are arranged and manipulated as tree structures to allow a compact representation of the data. The localization property of the wavelet transform is used to identify all wavelet coefficients that belong to a specific ROI within the volumetric data. The volume rendering operation is accomplished in the compressed domain thus reducing the computational time. This is possible because the wavelet transform and the projection are both linear operations that can be permuted. By considering all wavelet coefficients that contribute to the ROI from all resolution levels, a full resolution rendering of the ROI is obtained. The rendering process also uses the wavelet coefficients of the Context from a low-resolution level to achieve a multiresolution volume rendering. Our method allows the level of details outside the ROI to be progressively increased by incrementally adding the footprints from the details of the wavelet decomposition. Furthermore, in order to increase the visual quality of the rendered image without penalizing speed, we have used, in a preprocessing phase, a model that integrates opacity and improves shading.

In addition to demonstrating that the proposed method successfully achieved multiresolution volume rendering, we have discussed the optimal depth of the wavelet decomposition to attain best performance. We have also analyzed the impact of the ROI size on performance. We showed that, for a relatively small ROI, our multiresolution approach can improve the rendering speed considerably. A speed-up of a factor of 30 has been obtained using an ROI representing $10\% \times 10\% \times 10\%$ of the whole dataset.

The proposed method could be further improved. While already providing good performance and fast rendering, speed could be further improved using specialized graphics hardware. Our approach could be extended to store the footprints as textures, as proposed by Lippert [18], and to use the graphics hardware, as this has been proven to be very efficient [22].

Although the main objective in this paper was multiresolution volume rendering, rather than volume compression for efficient storage and rapid transfer time, our wavelet decomposition scheme, combined with the tree representation, are very useful to manage the data in a compressed and compact way, especially when dealing with large data sets. The octree provides satisfactory and efficient representation of the data. However other approaches based on quantization and

statistical compression methods could be explored in order to attain better compression rates. The impact of such approaches on image quality and decompressing speed would also need to be investigated.

## ACKNOWLEDGMENT

The authors would like to thank J. McDermott for reviewing the document.

## REFERENCES

[1] K. P. Andriole and R. L. Morin, "Transforming medical imaging: The first scar trip conference - a position paper from the scar trip subcommittee of the scar research and development committee," *Journal of Digital Imaging*, vol. 19, no. 1, pp. 6–16, January 2006.

[2] R. A. Drebin, L. Carpenter, and P. Hanrahan, "Volume rendering," in *SIGGRAPH '88: 15th Annual Conference on Computer Graphics and Interactive Techniques. Proceedings*. ACM Press, 1988, pp. 65–74.

[3] J. T. Kajiya and B. P. V. Herzen, "Ray tracing volume densities," in *SIGGRAPH '84: 11th annual conference on Computer graphics and interactive techniques. Proceedings*. ACM Press, 1984, pp. 165–174.

[4] M. Levoy, "Display of surfaces from volume data," *IEEE Computer Graphics Applications*, vol. 8, no. 3, pp. 29–37, 1988.

[5] O. Ratib, A. Rosset, M. Dahlbom, and J. Czernin, "Navigating the fifth dimension: new concepts in interactive multimodality and multidimensional image navigation," in *Medical Imaging 2005: PACS and Imaging Informatics*, vol. 5748. SPIE, 2005, pp. 28–32.

[6] H. L. Kundel, "Reader error, object recognition, and visual search," in *Medical Imaging 2004: Image Perception, Observer Performance, and Technology Assessment*, vol. 5372. SPIE, 2004, pp. 1–11.

[7] J. E. van der Heyden, K. Inkpen, M. S. Atkins, and M. S. T. Carpendale, "Exploring presentation methods for tomographic medical image viewing," *Artificial Intelligence in Medicine*, vol. 22, no. 2, pp. 89–109, 2001.

[8] P. Chang, J. Huffman, and B. Hebert, "Methods and apparatus for resolution independent image collaboration," United States Patent 65 567 24, 2003.

[9] P. Chang, J. Huffman, B. McCurtain, J.Reis, and B. Hebert, "Methods and apparatus for an image transfer object," United States Patent 69 382 11, 2005.

[10] O. Kuederle, K. Inkpen, M. S. Atkins, and M. S. T. Carpendale, "Interacting with image sequences: Detail-in-context and thumbnails," in *Graphics Interface*. Toronto, Ontario, Canada: Canadian Information Processing Society, 2001, pp. 111–118.

[11] E. C. Lamar, M. A. Duchaineau, B. Hamann, and K. I. Joy, "Multiresolution techniques for interactive texture-based volume visualization," in *Visual Data Exploration and Analysis VII. Proceedings*, vol. 3960, SPIE. Bellingham, Washington: The International Society for Optical Engineering, 2000, pp. 365–374.

[12] S. G. Mallat, "A theory for multiresolution signal decomposition — the wavelet representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, pp. 674–693, 1989.

[13] "Digital imaging and communications in medicine (DICOM). Supplement 106: Jpeg 2000 interactive protocol," 2006.

[14] M. A. Westenberg and J. B. T. M. Roerdink, "X-ray volume rendering by hierarchical wavelet splatting," in *15th International Conference on Pattern Recognition. Proceedings*, Barcelona, Spain, 2000, pp. 163–166.

[15] K. Krishnan, M. Marcellin, A. Bilgin, and M. Nadar, "Compression - decompression strategies for large volume medical imagery," in *SPIE Medical Imaging. Proceedings*, vol. 5371, February 2004, pp. 152–159.

[16] M. H. Gross, L. Lippert, R. Dittrich, and S. Häring, "Two methods for wavelet-based volume rendering," École polytechnique fédérale de Zurich, Tech. Rep., 1995.

[17] M. H. Gross, L. Lippert, A. Dreger, and R. Koch, "A new method to approximate the volume rendering equation using wavelets and piecewise polynomials," École polytechnique fédérale de Zurich, Tech. Rep., 1994.

[18] L. Lippert, "Wavelet-based volume rendering," Ph.D. thesis, Swiss Federal Institute of Technology, 1998.

[19] L. Lippert and M. H. Gross, "Fast wavelet based volume rendering by accumulation of transparent texture maps," *Computer Graphics Forum*, vol. 14, no. 3, pp. 431–44, 1995.

[20] S. Piccand, R. Noumeir, and E. Paquette, "Efficient visualization of volume data sets with region of interest and wavelets," in *Medical Imaging 2005: Visualization, Image-Guided Procedures, and Display*, vol. 5744. SPIE, 2005, pp. 462–470.

[21] S. Guthe, M. Wand, J. Gonser, and W. Strasser, "Interactive rendering of large volume data sets," in *VIS '02 : IEEE Visualization, 2002. Proceedings*, 2002, pp. 53–60.

[22] S. Guthe and W. Strasser, "Advanced techniques for high-quality multiresolution volume rendering," *Computers & Graphics*, vol. 28, no. 1, pp. 51–58, February 2004.

[23] J. Shapiro, "Embedded image coding using zerotrees of wavelet coefficients," *IEEE Transactions on Signal Processing*, vol. 41, no. 12, pp. 3445–3462, 1993.

[24] I. Gargantini, L. He, and Y. Zhou, "Comparison of wavelets for volume rendering," in *Ninth International Conference on Geometry and Graphics. Proceedings*, Johannesburg, 2000, pp. 68–73.

[25] S. Horbelt, M. Unser, and M. Vetterli, "Wavelet projections for volume rendering," in *Twentieth Annual Conference of the European Association for Computer Graphics (EUROGRAPHICS'99). Proceedings*, Milan, Italy, 1999, pp. 56–59.

[26] J. F. Blinn, "Models of light reflection for computer synthesized pictures," *Computer Graphics*, vol. 11, no. 2, pp. 192–198, July 1977.

[27] B. T. Phong, "Illumination for computer generated pictures," *Commun. ACM*, vol. 18, no. 6, pp. 311–317, 1975.

[28] M. Levoy, "Efficient ray tracing of volume data," *ACM Transactions On Graphics*, vol. 9, no. 3, pp. 245–261, 1990.

**Sébastien Piccand** Sébastien Piccand is a PhD student at the University of Limerick, Ireland. His research interests include sound and image processing, visualization and evolutionary learning.

He holds a Master of Telecommunication Engineering degree from the École Nationale Supérieure de Télécommunications de Bretagne (ENST Bretagne) in Brest, France, and a Master of Electrical Engineering degree from the École de Technologie Supérieure (ETS) in Montréal, Canada.

**Rita Noumeir** Rita Noumeir is a professor in the Electrical Engineering department at the Université du Québec, École de Technologie Supérieure in Montréal. Her main research interest is in the healthcare information technology, specifically, interoperability, electronic patient records, security, information confidentiality, image processing and information visualization.

She has been actively involved with Integrating the HealthCare Enterprise (IHE) since its inception, in establishing the broad direction and scope for the IHE demonstrations and in planning IHE workshops and other activities. She presented IHE at various international conferences and workshops, and participated in developing many IHE integration profiles that are detailed in the current IHE technical framework. She is a co-founder of IHE Canada.

Dr. Noumeir holds a Master's and Ph.D. degrees in Biomedical Engineering from the École Polytechnique, Université de Montréal.

**Eric Paquette** Eric Paquette is a professor at the École de Technologie Supérieure engineering school in the Software & IT Engineering department.

He is also a member of the LESIA research laboratory at the École de technologie supérieure. His research interests include computer graphics, realistic image synthesis and visualization.

Dr. Paquette received his B.Sc. in Computer Science from the Université de Sherbrooke, Canada, and his M.Sc. and Ph.D. degrees in Computer Science from the Université de Montréal, Canada. He is a member of the IEEE Computer Society and of ACM SIGGRAPH. He is also on the board of the ACM SIGGRAPH Montreal Chapter.