Integrating Sensors and Virtual Reality for Volumetric CT Analyses of Agricultural Soil Samples

Leonardo C. Botega[1,2,3], Paulo E. Cruvinel[1,2]
[1]Embrapa Instrumentation, São Carlos, SP, Brazil
[2]Post-Graduation Programs in Computer Science - Federal University of São Carlos, SP, Brazil
[3]São Paulo State University, Marilia, SP, Brazil
Emails: leonardo.botega@unesp.br, paulo.cruvinel@embrapa.br

Abstract - **Multi-modal sensing techniques and data fusion from sensors can offer new possibilities for providing agricultural soil analysis in a robust manner. In this paper we report the results of integrating X-ray Tomography (CT), a non-invasive sensing technology, within a Virtual Reality (VR) environment for agricultural soils analyses. In such a context, through a user interface, sensors, and a volumetric visualization of tomographic images a set of agricultural soils samples has been submitted for porosity analyses. The use of graphic computational resources allowed the addition of functionalities, like volumetric visualization and immersion. For validation, it has been used a case study, involving analysis of porosity of agricultural soils samples. In fact, using energy of 59.6 keV and time window equal to 10 seconds for sampling of each tomographic projection it has been possible to reconstruct digital tomographic images from agricultural soils to be analyzed in such a system. Results indicate both the preferential paths for the water flow and a new way for evaluation of the physical properties of an agricultural soil.**

Keywords - X-ray sensors; virtual reality sensors; digital image processing; X-ray tomography; agricultural soil porosity; decision-making process.

I. INTRODUCTION

Direct and indirect measurements can be used to evaluate physical, chemical, and biological inputs availability in agricultural soils. In fact, both are based on the use of sensors. However, when there are needs for the spatial variability evaluation of those variables into agricultural soils, not only sensors but also methods should be taken into account. In fact, sensors and methods should be integrated to allow decision making related to the agricultural production processes [1].

Besides, evaluating the evolution that is happening in the soil science area, it is noticed an increasing interest of the scientific community in the development and application of non-invasive techniques for the study of physical characteristics of agricultural soils.

In such a context, since the 1980 decade, the application of image sensors based on Computed Tomography (CT) for agricultural soils imaging [2]–[9] has become one of the noninvasive methods for the evaluation of the water movement into soils due to morphology, and aggregates distribution. In fact, such kind of instrumental arrangement has provided improvements in relation to those techniques based on the use of gravimetric and neutron probe for water content measurements in agricultural soils [10][11].

Additionally, combined with the development of CT, new methods of three-dimensional (3-D) reconstruction were developed, mainly motivated by the lack of information from two-dimensional models for a precise diagnosis in studies that require volumetric information [12]. Another challenges regarding to such aspects were associated with the image reconstruction process, as well as those related to the reconstruction algorithms, the computational capacity, and the way to handle large amounts of data [13]. Therefore, since that time, it has been understood that working with tomographic reconstruction implied to take into account a large amounts of data and the need to have available a large processing capacity [14][15].

Moreover, due to the advent of precision agriculture, it had become imperative to have adequately models for management based on data analyses not only related to the spatial variability but also the temporal one in the areas used for agriculture.

In this sense, the standardization of data storage and the architecture of distributed information systems that allow integration of different types of data in a simple and transparent way had become to be quite important for the development of new methods for non-invasive analyses in agricultural industry [16]-[21].

Into such a subject, as an example, digital agricultural soil images are obtained by tomography taking into account several projections. Moreover, because one soil sample is scanned at different angles, a large amount of data should be computationally processed. Nowadays, the use of tomography not only allows us to obtain information about soil density and moisture at the pixel level but also allows quantification of the pore volume and its representation in three dimensions. The soil pores vary in size and shape and can be interconnected.

In 1982, Bouma has highlighted the importance to determine the continuity of the pore network for the flow of water in soil [22]. Therefore, not only pore diameter but also pore continuity interferes in the process of redistribution of soil water. In such a context, it is important to assess the porosity of the soil, because, depending on the soil management strategy adopted for planting, restriction of soil water flow may occur, thus compromising plant growth. To determine the soil porosity, volumetric measurements are conventionally used [23][24]. For this, it is necessary to collect undisturbed soil samples for quantitative evaluation of its porosity based on the use of tomographic scanners.

Besides, methods based on volumetric reconstruction have been developed for such a purpose, mainly due to the inadequacy of information of two-dimensional models for accurate diagnosis, in studies that need volumetric information. Thus, such methods suggest the composition of surfaces and volume of the samples under analyses, i.e., contributing with the increase of precision in process of information extraction. However, it is still a challenge gathering all the information from the agricultural soils, i.e., the continuity, size, and shapes of the pores in a soil sample, among others.

CT is one methodology that allows observing the structural components of the soil, allowing better visualization of the behavior of the structure and soil porous space. A bi-dimensional CT image indicates the amount of radiation absorbed by each portion of an analyzed sample. In fact, the amount of the radiation absorption can be associated to a calibrated scale.

Since the X-ray absorption capacity of a material is closely related to its density, different density areas can be represented by either pseudo-colors or by a gray tone values. Therefore, based on the intensity emitted by an x-ray source and the intensity captured by the detector at the other side of the propagation line, one can determine the attenuation weight due to the object that is located between the source and the detector. The data related with the attenuations and their weights are crucial for the reconstruction process from projections (Figure 1), which allows mapping all the linear attenuation coefficients into a slice of the sample.

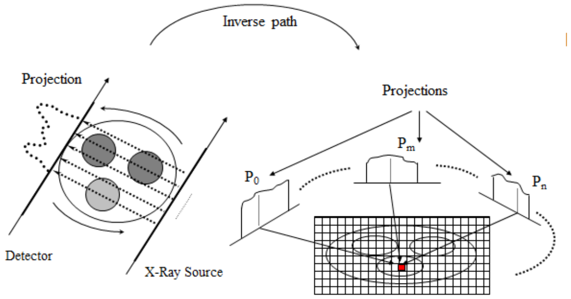


Figure 1. General view of the CT image reconstruction from projections
.

The calculation of the attenuated photons intensity in relation of the initial photons intensity can be obtained as follows:

$$N = N_0 e^{\oint_p (-\mu(\rho, Z_N, E)x)dp} \qquad (1)$$

where (N) is the attenuated photons number, (N_o) is the initial photons intensity, (μ) is the linear attenuation coefficient (in cm^{-1}), (ρ) is the material density (g/cm^3), (Z_N) is the atomic number of the material, and (E) is the X-ray energy.

In addition, if the study sample is a chemical component or a mixture, like an agricultural soil, then its mass attenuation coefficient can be roughly evaluated based on the linear attenuation coefficients of each element. Furthermore, the final mass attenuation coefficients can be mapping, i.e., taking into account the spatial variability of the pixels, whose intensities can be given by:

$$\frac{\mu}{\rho} = \sum_i w_i \left(\frac{\mu_i}{\rho_i} \right) \qquad (2)$$

where (w_i) is proportional to the weight of the i_{th} constituent of the sample`s material.

However, the interconnection for preferential flow requires additional methods, which can be beyond its use.

Besides, such an innovation can be faced by taking into account the composition of CT with sensors-based VR techniques, to assist noninvasive research through immersive and interactive processes.

The VR was born in the eighties under the need of differentiating traditional computational simulations of the synthetic worlds that began to stand out. This initiative gave credit to researchers like Bolt [25] and Lanier [26]. VR transports the individual into a fully immersive and interactive experience with a degree of realism. Likewise, academics, software developers and researchers have been still looking for defining a VR based on their own experiences. However, it is possible to observe in specialized literature that all of them technically considered the term related to the immersive and interactive experience, i.e., based on images generated by computers, rendering or not in real time [27]-[30]. Furthermore, the use of sensors in their external devices, i.e., digital gloves, video-helmets, digital caves, digital tables, among other, led to the concept related to sensors-based VR.

In 1994, Machover has stated that the quality of a VR system is essential, because it stimulates to the maximum the user, in a creative and productive way, providing feedbacks in a coherent way to the user's movements [31].

Until the present moment, just some units of research have been developing projects using sensors-based VR applications in the area of scientific visualization, as the tomographic reconstruction, due to the high cost and to technical difficulties involved in such processes. However, some proposals have been appearing to minimize the difficulties of development and maintenance of the systems and necessary programs.

Additionally, today a better organization of human resources is being observed to integrate areas of the knowledge leading to the application of such advanced methods based on the connection and use of those technologies.

Thereby, the main objective of this work is to present the development of a VR system to support the analysis of 3D reconstructed soil samples using innovative immersive visualization and interaction techniques by integrating sophisticated external sensor-based devices.

Specifically, it is presented the organization and implementation of a synthetic environment, which makes possible the visualization, analysis and manipulation of soil samples produced by an algorithm of volumetric reconstruction of X-ray tomographic images, through graphic computational tools and non-conventional sensor based-VR devices, aiming immersion and user interaction to the scene entities, making possible the non-destructive analysis of agricultural soil samples, as shown in a case study in Soil Science.

The remainder of the paper is organized as follows: Section II presents the materials and methods; Section III presents the results, discussions, and performance evaluation; finally, conclusion and future work are presented in Section IV,

II. Materials and Methods

The conceptual and methodological structuring applied in the development of the sensors-based VR system dedicated to the inspection of digital tomographic images from agricultural soils, uses data obtained by means of a volumetric reconstruction algorithm. Figure 2 shows a general view of the sensors-based VR system dedicated to the tomographic inspection of agricultural soil samples, as well as the dataflow, where, from tomographic image data, such soil samples can be reconstructed, imported and treated by several VR processes, i.e., focusing analyses related to the soil science area.

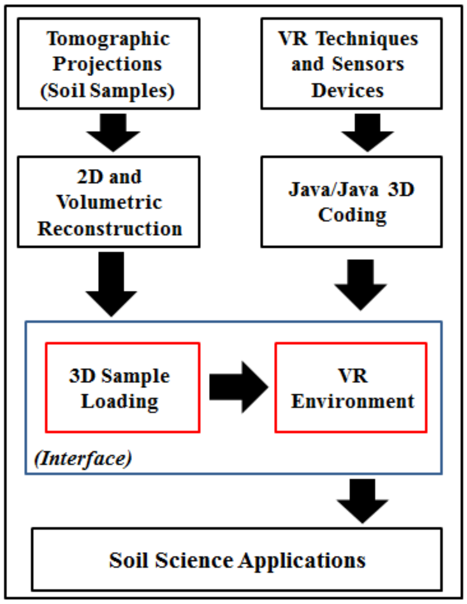

Tomographic Projections (Soil Samples)
VR Techniques and Sensors Devices
2D and Volumetric Reconstruction
Java/Java 3D Coding
3D Sample Loading
VR Environment
(Interface)
Soil Science Applications

Figure 2. General view of the sensors-based VR system customized to the inspection of tomographic samples of agricultural soils, as well as a view of the dataflow from acquisition to the visualization process.

The software system was organized taking into account the concept of classes. In object-oriented programming, a class is an extensible program-code-template for creating objects, providing initial values for state (member variables) and implementations of behavior (member functions or methods). In this work, the following classes have been considered: *Reconstruction*; *Loader*; *Transformations*; *Polygonal Attributes Extraction*; *Filter*; *Transparency*; *Illumination*; *Coloring*; *Conventional Collision*; *Non-conventional Collision*; *Conventional Model Manipulation*; *Non-conventional Model Manipulation*; *Conventional Scene Manipulation*; *Non-conventional Scene Manipulation*; *Quaternion*; *Visualization*; and *VR Environment*.

All devices were implemented using the Java programming language and the *Java3D* API [32].

For the obtaining of the tomographic image data, the CT scanner from Embrapa Instrumentation was used. All the tomographic projections allowed the images reconstruction, turning possible the generation of mass attenuation coefficient maps, i.e., given in [cm^2/g], with spatial resolution equals or larger than 1 mm. All the soil samples were submitted to the acquisition process under energy of 59.6 keV and time window equal to 10 seconds for sampling of the points for the tomographic projection.

For two-dimensional reconstruction, it was used an algorithm of *Filtered Back-Projection* (FBP), with a filtering based on the use of the *Hamming´s* window, implemented under 1-D *Fast Fourier Transform* (FFT), using the C++ language [33]. After that, with the 2-D reconstructed images, a suitable filtering technique was also used. Such a filtering technique was based on the use of *Wavelet Daubechies Transform* (WDT), which allowed filtering only certain image areas preserving borders and details, i.e., through using a window with 76 coefficients [34].

For the volumetric reconstruction it was adopted an interpolation based-overlapping algorithm of reconstructed two-dimensional slices. Such a technique consists of setting up the plans generated by the functions f(x, y, z_i) for i = 0...(n-1), where n is the number of reconstructed plans. Consequently, specific two-dimensional slices were interpolated to reconstitute the spaces left among these overlapped plans.

Figure 3 shows the original plans overlapping and the interpolated plans. Such a method was used to reduce the computational costs and the radiation time, based on the use of interpolation in between the spaces of the reconstructed slices based on the use of *B-splines* [35]. Thus, with only a few slices, the algorithm was prepared to estimate and complete the entire information.

Besides, the sensors-based VR system for the inspection of agricultural soils samples was organized taking into account the CT images, and a set of non-conventional sensors to support the *VR environment*.

In addition, for the evaluation of the preferential paths for the water movement in soil, sensors were used to detect the motion based on the use of gloves, the space based on 3-D visualization, i.e., using a CCD *head-mounted display*, as well as microelectromechanical actuators based on piezo-electrical devices [36][37].

Figure 4 shows photos of the used CCD glasses with sensors, model GSD 300 from Innovatek^{TM}, which has been used in the method for allowing the virtual reality including headsets sensors for properly align with the screen of the computational environment area, in order to reduce distortions.

Such sensors were necessary to translate the movement and to help the understanding of users in relation of the workspace into the agricultural soil samples. At the end of the process, the volumetric model is converted into *Wavefront File Format* (.obj), i.e., using the *vtkOBJExporter* class from *vtkOBJExporter.h* package of the visualization toolkit. This format has been chosen for its high performance and flexibility on import such models to virtual environment, where all their attributes can be customized for graphic API's.

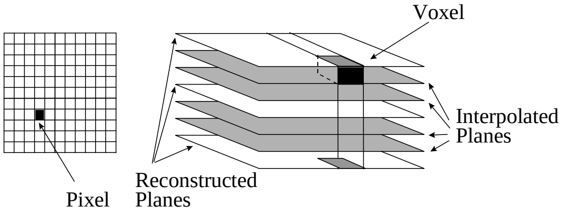
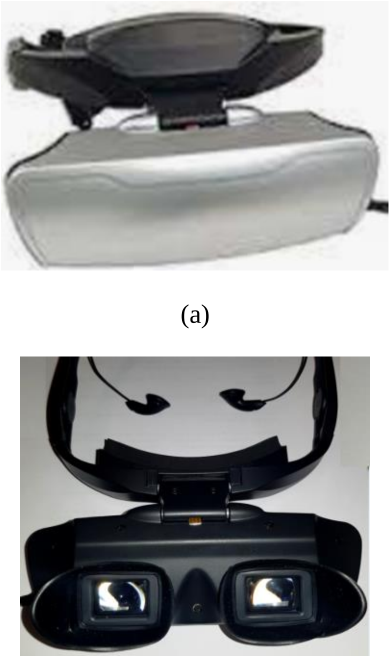


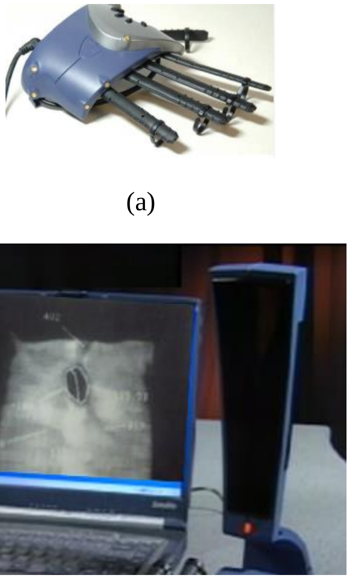
Figure 3. Volumetric reconstruction based on a set of reconstructed slices and the use of B-spline interpolator.


(a)

(b)

Figure 4. Details of the used virtual glasses based on sensors devices. In (a) the frontal mechanical view, and in (b) the CCD head-mounted display, microelectromechanical actuators, and headphones.

Figure 5 shows photos of the P5Glove with sensors obtained from the Mindflux^{TM}, which has been used for the 3-D virtual controller system. It is ergonomically adequate designed to allow for comfort during use. The glove features an infrared control receptor with an anti-reflective and scratch resistant lens. The main characteristics of the P5 glove includes a virtual 3D controller; Mouse-mode compatibility; 6 degrees of tracking (X, Y, Z, Yaw, Pitch and Roll) to ensure realistic movement; bend-sensor and optical-tracking technology to provide true-to-life mobility; as well as plug-and-play setup using an Universal Serial Bus (USB) port from a computational system.


(a)

(b)

Figure 5. (a) The P5Glove used for the 3-D virtual controller system; (b) the P5Glove`s control tower connected in a computer, i.e., based on infrared receptor with an anti-reflective lens.

The *Attributes Extraction* class treats of obtaining the voxels data from a volumetric image, using those above mentioned input non-conventional devices, i.e., supplying to the users' information on a specific point of the volumetric representation.

Initially, the objects of the classes *PickCanvas* and *PickResult* are instantiated, and these objects are responsible for activating the data extraction of a *Canvas3D* object and storing such data in vectors of events results.

Based on the user interest a region can be selected and attributes can be extracted using a coordinate z, since it can be stabilized on the selected region in the display, allowing selection through a two-dimensional viewport in an intuitive way.

Thus, the available data for picking operations under instances of Shape3D and their respective methods are: the borders, with *getBounds*; the scene graphs, with *getLocale* and *numBranchGraph*; the geometries, with *getGeometry*; *ColoringAttributes*, with *get.ColoringAttributes*; the material under the Hue, Saturation, Lightness (HSL) and Red, Green, Blue (RGB) formats, with *getMaterial*; the *transparency,getTransparency*; and the polygons, with the *getAppearance.getPolygonAttributes.getPolygonMod* class.

In addition, an object is instantiated, belonging to the *PickIntersection* class, also of the *com.sun.j3d.utils.picking* package, responsible for sheltering the collision point between an entity/node and the two-dimensional cursor. Thus, this instance stores in its content the intersection product among an entity of *PickResult* with the chosen Canvas3D point, which passed to the *getClosestIntersection* method as parameter. Like that, the *PickIntersection* class can offer through its events: the distance between the point and the observer, with the *getDistance* method; the coordinates of the point, with the *getCoordinates* method; the coordinates of the closest vertex, with the *getClosestVertexCoordinates* method; the normal straight line of the point, with the *getNormal* method; and the transformation head offices with the *getMatrix* method.

Besides, the classes *PickIntersection* and *PickResult*, as well as the *Attributes Extraction class* can allow the reading of each mass attenuation coefficient values, which are present in the tomographic volume. In this context, these values can be obtained through the gray level tones, which are represented by luminance, index "L" from the HSL pattern, obtained by using the class *getMaterial* method.

The *Non-conventional Scene Manipulation* class is one of the most important for the user interactivity and immersion in the *VR environment*, once it allows the user browsing in all directions through the synthetic scene, approximating and going into the reconstructed structures using the data gloves *P5Glove* class [38].

For the accomplishment of such events, the manipulation classes and the model of the scene are both based on another auxiliary class called *FPSGlove*, which is available in the *com.essentialreality* package offered by the devices manufacturer. The *FPSGlove* class is responsible for including all the parameters regarding the non-conventional devices, concerning to the positioning, orientation and fingers bending, making possible to detect the proximity and inclination on it, and thus launches a series of customized events.

On the other hand, in relation of the constructor method of classes, additional parameters of same importance can be activated, such as: (1) *P5_Init*; (2) *P5_setForward*; (3) *P5_setMouseState*; (4) *P5_setFilterAmount*; and (5) *P5_setRequiredAccuracy*. These classes are responsible for initializing, determining the positive direction, and turning off the mouse, filtering the sign and determining the precision movements, respectively. Soon afterwards, the methods responsible for detecting the position of the glove in the real environment are declared. The methods are the *getXPosition, getYPosition* and *getZPosition,* which map the triggers mentioned before to launch an event type, it means, they monitor the values received by the glove through instances of the class *P5State*, a class responsible for determining the current state of the glove. Thus, through the *filterPos* method of *P5State,* the exact position of the device is obtained and then assigned to the methods to check if the limits were or not outdated.

In a similar way to the positioning detection methods, still in the *FPSGlove* class, the methods *getYaw*, *getPitch* and *getRoll* are described, which answer for detecting the inclination of the device in the axis Y, X, and Z, determining if the established limits for the flags were reached.

After having implemented the monitors and triggers of events with the auxiliary class, the *Non-conventional Scene Manipulation* should now have their events described in its scope.

For such a scope, firstly the used class should be extended to the *ViewingPlatform* class, it means, to have their instances interpreted as events on the virtual scene. Besides, two other specific parameters are included, the translation step and the rotation one, which are responsible for defining when the virtual models will be moved or lean in each movement of the device in the real world, after being recognized by the *FPSGlove* class.

Once such a process is concluded, the instance of the *Non-conventional Scene Manipulation* class should be harnessed to the object of the ViewingPlatform class of the current *Canvas3D* object, so that all of the movements can be relationated on the scene and not the volumetric model, i.e., through the *setViewingPlatform* method.

The *Non-conventional Model Manipulation* is a class responsible for accomplishing the three-dimensional representation movements through real movements of the data glove *P5Glove*, where the user can change the positioning and orientation of models in real time in all directions and angles, contributing to the virtual reality environment interactivity in six degrees of freedom. To operate such a process, it is important to take into account the *Non-conventional Scene Manipulation*, in which the current implementation uses the support *FPSGlove* class.

Thus, the *Non-conventional Model Manipulation* class is extended of the Behavior class, a class that describes behaviors, customized for reactions to the movements of the device.

Besides, after assigning the procedures *getXPosition*, *getYPosition*, and *getZPosition* to obtain the positions, as well as the procedures *getYaw*, *getPitch*, and *getRoll* to obtain the orientations, under instances of the *FPSGlove* class, the procedure *rotateQuaternion* is called. Such an operation is based on the transformation of the Euler angles in Quaternion coordinates, i.e., useful for establishing the rotations using complex numbers and imaginary axis to improve the movements precision [39]. A Quaternion (q) is represented by:

$$q = \left(s,\ \vec{v}\right) \tag{3}$$

where the (s) and ($\vec{v}$) are representing the real and vectorial components respectively.

The relevance of using Quaternions is related with the opportunity for applying rotations in data collected from sensors, i.e., using three-dimensional models, based on the use of vectorial products. For the *Quaternium* class implementation the Java programming language has been used (Java3D API).

For the Quaternion`s model, we have divided the procedure in three main steps: (1) step for mapping the position and orientation of the glove, i.e., based on the data collected from positioning sensors, (2) step for coordinates conversion; and (3) step for the parameterization to allow rotation and visualization based on synthetic scenes entities.

The P5Glove used is an unconventional device having only 128 g. It is mouse compatible during operation. It also presents fold sensors, which are located on the fingers structure, i.e., being responsible for the identification of the movements, as well as the actions for holding a sample in the synthetic RV environment. Such sensors can have their parameters customized through the use of an appropriate API, i.e., called *Dualmode*. As observed previously in the technical features descriptions, its operation is based on an optical tracking system and two photosensitive receivers located in a mechanical tower. Also, to perform the positioning data mapping and orientation the glove has been used taking into account all the available six degrees of freedom. Likewise, an additional class has been developed to handle the signals from the glove`s hardware, interpreting the drivers provided by the manufacturer.

The coordinate's conversion takes place according to the Pseudocode 1 (Figure 6), which provides not only the scalar but also the vectorial part of the Quaternion. The Pseudocode 2 (Figure 7) presents the application of the Quaternion terms to the matrices transformation.

```
Begin
    $New\,rotation = \sin(angularStep/2)$
    $q_1 \cdot x = rotation * gloveAxis_1$
    $q_1 \cdot y = rotation * gloveAxis_2$
    $q_1 \cdot z = rotation * gloveAxis_3$
    $q_1 \cdot w = \cos(angularStep/2)$
End
```

Figure 6. Pseudocode 1, which is part of the code to convert from Euler to Quaternion.

```
Begin
    $transformationMatrix[0] = (1.0 - 2.0 * q_1 \cdot y * q_1 \cdot y - 2.0 * q_1 \cdot z * q_1 \cdot z) * scale[0]$
    $transformationMatrix[4] = (2.0 * (q_1 \cdot x * q_1 \cdot y + q_1 \cdot w * q_1 \cdot z)) * scale[0]$
    $transformationMatrix[8] = (2.0 * (q_1 \cdot x * q_1 \cdot z - q_1 \cdot w * q_1 \cdot y)) * scale[0]$
    $transformationMatrix[1] = (2.0 * (q_1 \cdot x * q_1 \cdot y + q_1 \cdot w * q_1 \cdot z)) * scale[1]$
    $transformationMatrix[5] = (1.0 - 2.0 * q_1 \cdot x * q_1 \cdot x - 2.0 * q_1 \cdot z * q_1 \cdot z) * scale[1]$
    $transformationMatrix[9] = (2.0 * (q_1 \cdot y * q_1 \cdot z - q_1 \cdot w * q_1 \cdot x)) * scale[1]$
    $transformationMatrix[2] = (2.0 * (q_1 \cdot x * q_1 \cdot z - q_1 \cdot w * q_1 \cdot y)) * scale[2]$
    $transformationMatrix[6] = (2.0 * (q_1 \cdot y * q_1 \cdot z + q_1 \cdot w * q_1 \cdot x)) * scale[2]$
    $transformationMatrix[10] = (1.0 - 2.0 * q_1 \cdot x * q_1 \cdot x - 2.0 * q_1 \cdot y * q_1 \cdot y) * scale[2]$
End
```

Figure 7. Pseudocode 2, which is part of the code dedicated to apply the Quaternion for the matrices transformation.

Hence, back to the main process, the *rotateQuaternion* procedure assigns to its class the axis and angles parameters, in Euler coordinates and returns a Quaternion description, a set used in the same *Quat4f* constructor, constructing a Quaternion of float, and after in setRotation executing a rotation with instances of *Transform3D*.

The *Quaternion* class implements a conversion algorithm so the system stops using just rotations on the axis x, y and z, and starts to accomplish orientations on some intermediate axis, defined by a vector that goes through the origin and reaches a point in the space. Such kind of an axis can be represented by a specific coordinate of the real device, e.g., the Cartesian coordinates (x, y, z) of one of the eight LED's present in the controller tower, which is used with the glove.

To accomplish this operation, imaginary bases and complex numbers are used, providing an alternative parameter for *setRotation*, method of the *Transform3D* class, which allows using a Quaternion as an argument. Thus, calling an instance of *Quaternion* to accomplish a rotation with the non-conventional device *P5Glove*, the orientation of the glove is interpreted by the *FPSGlove* class and translated by *Non-conventional Scene Manipulation* or *Model*, is converted from the Euler system to *Quaternion* base, returning the system new orientation coordinates, to be executed by the *Quat4f* method of *Transform3D*, which encapsulates the whole functioning of the Quaternion previously described.

At the end of the process, the product of *Non-conventional Model Manipulation* class is encapsulated in a *BranchGroup* object and assigned to the transformation group, *TransformGroup*, which conducts the three-dimensional representation movements, in a distinct way of the previous class. In such a way, not only all the movements' detection, but also the effective positioning change, and the entities orientation produce effects under the current models in the *Canvas3D* object.

The *Non-conventional Collision* class treats the implementation of a collision detection algorithm added to the *Non-conventional Scene Manipulation* class, restricted to events that use non-conventional input data devices, specifically the data glove *P5Glove*. In that way, through the algorithm, the users are also prevented to cross the faces of a three-dimensional representation during the browsing process in synthetic scenes, allowing only the cameras transpositions inside the empty spaces among such faces, simulating real physical processes.

Thus, each spatial position of the glove is tested as the current instance of itself, it means, to each direction of movements in some specific moment, where the possible alternatives are: left, right, up, down, forward, and back. After identified the positioning of the glove, in the moment of a supposed collision, the *Non-conventional Collision* class can blocks the device movements. Thus, the last movement of the glove when colliding is stopped, although the data glove can freely be moved in the real environment. This is caused by a new instantiation of the current positions of the glove, assigning to them, empty vectors, in other words, initialized in the origin, i.e., causing the immediate stop of the device movements.

Additionally, at the same time, when accomplishing any other move, which does not take them to a continuation of the blocking, the class interprets them and allows continuing the valid movements series, through a new instantiation of the mapped positions of the glove, using as parameter the position where the collision was begun and the linear step adopted by the class. At the end of this process, a *Shape3D* is added to *PhysicalBody* to detect in the browsing the scene being used by a user, allowing interaction and selection of each three-dimensional faces. Besides, the algorithm of such a class allows both preventing the browsing to continue or not in a scene, as well as an information of the direction of the glove movement, since active.

For the implementation of the *Visualization* class, the system interface prepares a volumetric tomographic image to be visualized. In such a way, the volumetric tomographic image is prepared to be adjusting to the 3-D model, i.e., occupying the whole extension of the *Canvas3D* object. Such activity contributes to improve the visualization quality, once the resolution of the HMD screens is inferior to the conventional LCD and CRT monitors.

III. RESULTS AND DISCUSSIONS

Based on the use of the tomographic projections and the two-dimensional reconstruction FPB algorithm it was possible to get volumetric images by means of the *B-spline* use. Figure 8 presents a set of examples related to the volumetric tomographic images obtained for stratified agricultural soil, degraded soil and a clay soil sample, respectively.

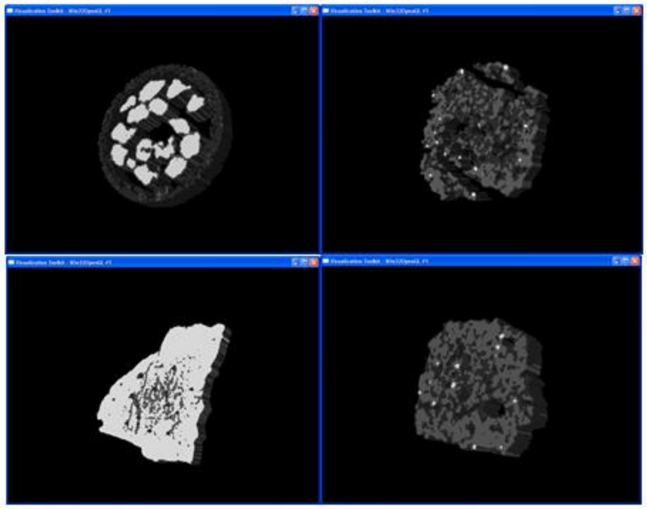


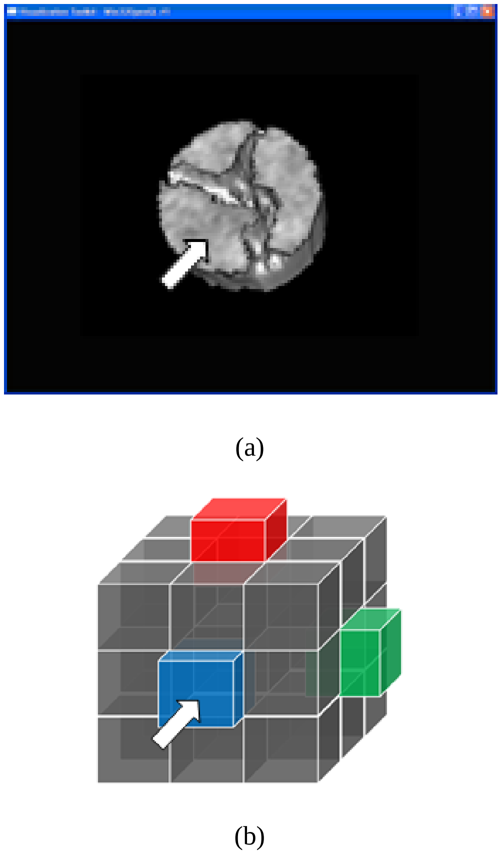
Figure 8. Volumetric images reconstructed by FBP, and the B-Spline algorithm

Based on the *Attributes Extraction* class, intrinsic characteristics of the scene and of agricultural samples could be obtained, through the use of either the mouse or the P5Glove, having as data origin the three-dimensional representations in the VR environment. Such a class allows the users to select any voxel from a set of voxels, which is useful for the 3D soil samples analysis during the information retrieval process. The set of data was divided into 2 categories: one of then concerning to the Scene, and other concerning to the CT measurements.

In relation to the first category, the synthetic scene data have been related to: borders, which have represented the geometry limits or the geometry limits that involved it; the scene graph, which has represented the node hierarchy in the tree; the current geometry in the model and its composition; the distance of a certain voxel in relation to the coordinates chosen in the scene; the closest vertex to the chosen point in the scene; the three-dimensional coordinates; and the normal straight line in the closest face, which had involved the chosen coordinates.

Secondly, concerning the tomographic data, the obtained data were: color attributes, which had represented the individual color of each voxel, independent of illumination intensity; the mass attenuation coefficients values of the agricultural soils, which are represented by the colors of each analyzed voxel, i.e., related to the light intensity in each position; transparency attributes; polygons attributes, and finally the saturation and matrix, of the HSL coefficients.

An experiment for validation of the result was prepared taking into account a digital and volumetric tomographic image obtained from a latosol soil. For such volumetric image, the value of an arbitrary voxel was taken as presented in Figure 9.



(a)

(b)

Figure 9. a) Volumetric image from a latosol sample, where an arbitrary point is chosen using a conventional device (mouse) or a non-conventional one (glove). The coordinates of the voxel are directly selected and respective information can be exhibited for users; b) Voxel selection from a 3D representation, i.e, illustrating the use of the *Attributes Extraction* class.

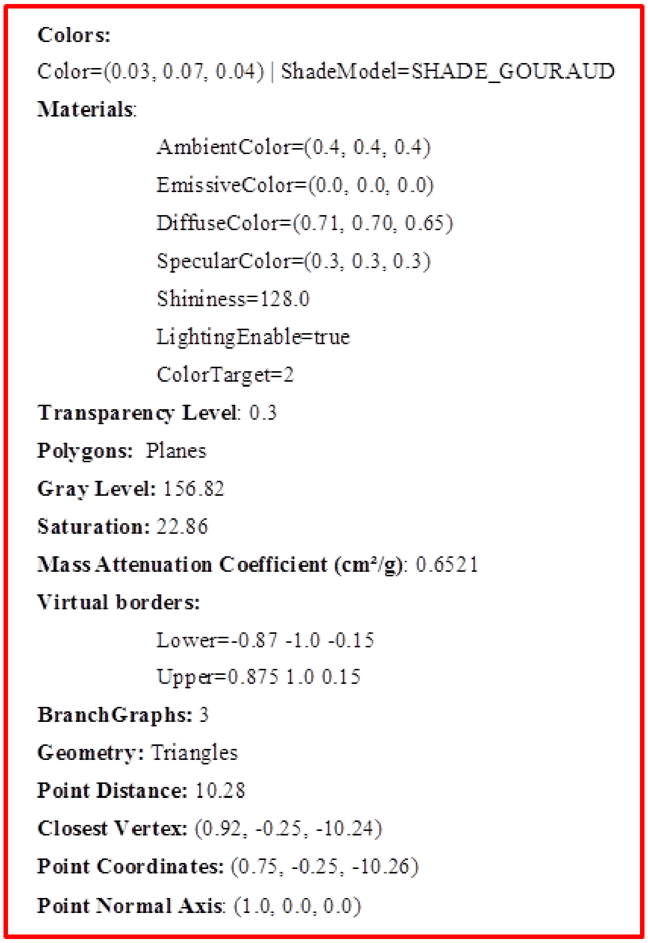
The developed method has allowed obtaining from a latosol samples sets of attributes, i.e., by assuring the reliable recovery of the samples agricultural data through the choice of voxels by selecting their coordinates (Figure 10).

In relation to the use of the *Non-conventional Scene Manipulation* Class, it has been possible to observe it usefulness to control the P5Glove, i.e., when one is browsing in a scene. In this context, according to this device position, users can be browsing through the scene, where the hand`s displacement can be faithfully translated to the scene`s movements in real time, even when the RV environment's cameras are moving. In an analogous way, such movements are also translated into the three-dimensional (3D) displacements. The *Non-conventional Model Manipulation* class simulates the manual support of 3-D samples, as well as its total movement inside the scene, with 6 degrees of freedom.

The *Quaternion* class has presented an adequate operation, making possible the correct conversion from the Euler`s coordinates to the Quaternion`s coordinates. Furthermore, the result obtained with the application of the Transform3D class has produced a smooth orientation changes for agricultural soil analysis.

Additional examples of results are presented below, i.e., considering an angular sector equal to 180º, divided into subsectors of 45º. For such examples, a set of voxels has been initially positioned at the coordinate (0.0, 0.0, 0.0).

Besides, the voxels have been rotated around an imaginal line, which has been represented by a dotted line. Moreover, it has been considered that such an imaginal line has both passed through the origin coordinate and reached the geographical position of three of the eight LEDs located on the control tower of the data relate to the glove.



Colors:

Color=(0.03, 0.07, 0.04) | ShadeModel=SHADE_GOURAUD

Materials:

AmbientColor=(0.4, 0.4, 0.4)

EmissiveColor=(0.0, 0.0, 0.0)

DiffuseColor=(0.71, 0.70, 0.65)

SpecularColor=(0.3, 0.3, 0.3)

Shininess=128.0

LightingEnable=true

ColorTarget=2

Transparency Level: 0.3

Polygons: Planes

Gray Level: 156.82

Saturation: 22.86

Mass Attenuation Coefficient (cm^2/g): 0.6521

Virtual borders:

Lower=-0.87 -1.0 -0.15

Upper=0.875 1.0 0.15

BranchGraphs: 3

Geometry: Triangles

Point Distance: 10.28

Closest Vertex: (0.92, -0.25, -10.24)

Point Coordinates: (0.75, -0.25, -10.26)

Point Normal Axis: (1.0, 0.0, 0.0)

Figure 10. Sets of attributes and data from voxels by selecting their coordinates.

Table I, as well as Table II, and Table III, are showing results related to the intermediate rotations that are respectively associated with the following operations:

i. Dotted line segment, between the coordinates (-1.0, 1.0, 0.0) from the tower of LEDs and (0.0, 0.0, 0.0) from the origin (Figure 11);

ii. Dotted line segment, between the coordinates (1.0, 1.0, 0.0) from the tower of LEDs and (0.0,0.0,0.0) from the origin (Figure 12);

iii. Dotted line segment, between the coordinates (0.0, 1.0, 0.0) from the tower of LEDs and (0.0, 0.0, 0.0) from the origin (Figure 13).

TABLE I. RESULT OF THE ROTATION OF 180º, CONSIDERING THE GEOGRAPHICAL POSITION OF THE LEDS IN THE COORDINATE (-1.0, 1.0, 0.0).

Sample's Initial Position	Quaternions	Sample's Final Position
(0.0, 0.0, 0.0)	vector: (0.0, 0.0, 0.0), scalar: 0.92	(-0.70, 0.70, 0.0)
(-0.70, 0.70,0.0)	vector: (-0.27, 0.27, 0.0), scalar: 0.92	(-0.99, 0.99, 0.0)
(-0.99, 0.99, 0.0)	vector: (-0.38, 0.38, 0.0), scalar: 0.92	(-1.29, 1.29, 0.0)
(-1.29, 1.29, 0.0)	vector: (-0,49, 0,49, 0.0), scalar: 0.92	(-1.68, 1.68, 0.0)

TABLE II. RESULT OF THE ROTATION OF 180º, CONSIDERING THE GEOGRAPHICAL POSITION OF THE LEDS IN THE COORDINATE (1.0, 1.0, 0.0).

Sample's Initial Position	Quaternions	Sample's Final Position
(0.0, 0.0, 0.0)	vector: (0.0, 0.0, 0.0), scalar: 0.92	(0.70, 0.70, 0.0)
(0.70, 0.70, 0.0)	vector: (0.27, 0.27, 0.0), scalar: 0.92	(0.99, 0.99, 0.0)
(0.99, 0.99, 0.0)	vector: (0.38, 0.38, 0.0), scalar: 0.92	(1.29, 1.29, 0.0)
(1.29, 1.29, 0.0)	vector: (0.49, 0.49, 0.0), scalar: 0.92	(1.68, 1.68, 0.0)

TABLE III. RESULT OF THE ROTATION OF 180º, CONSIDERING THE GEOGRAPHICAL POSITION OF THE LEDS IN THE COORDINATE (0.0, 1.0, 0.0).

Sample's Initial Position	Quaternions	Sample's Final Position
(0.0, 0.0, 0.0)	vector: (0.0, 0.0, 0.0), scalar: 0.92	(0.0, 0.70, 0.0)
(0.0, 0.70, 0.0)	vector: (0.0, 0.27, 0.0), scalar: 0.92	(0.0, 0.85, 0.0)
(0.0, 0.85, 0.0)	vector: (0.0, 0.32, 0.0), scalar: 0.92	(0.0, 0.92, 0.0)
(0.0, 0.92, 0.0)	vector: (0.0, 0.35, 0.0), scalar: 0.92	(0.0, 0.95, 0.0)

Such examples of results are related to the operations that have allowed finding the new Quaternions. For them, the real part is resulting of the calculation from the cosine of the selected rotation angle. The imaginary part allows the evaluation of the directions of these new Quaternions, i.e., in relation to the reference axes. In fact, to use the approach presented in the examples above, the angular rotation step should be smaller than 180º. Actually, this represented constrains to avoid failures in finding the direction of the Quaternion. Nevertheless, a reliable operation has been obtained including calculation not only for the direction but also for the orientation of the Quaternion, as shown in Figure 14. Furthermore, by doing such operation no losses in relation to the degrees of freedom have occurred.

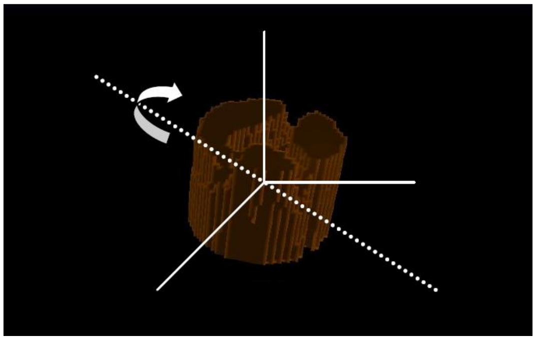


Figure 11. Result showing the rotation around the dotted line defined between the geographical position of the LEDs in the coordinate (-1.0, 1.0, 0.0) and the origin of the reference axes.

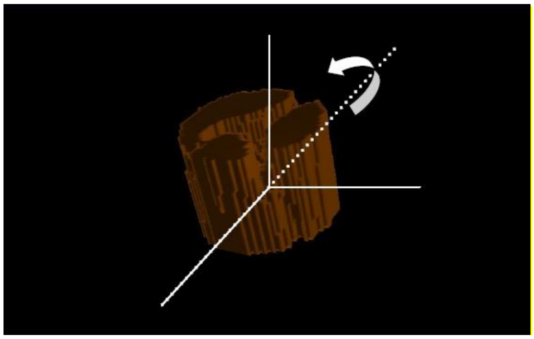


Figure 12. Result showing the rotation around the dotted line defined between the geographical position of the LEDs in the coordinate (1.0, 1.0, 0.0) and the origin of the reference axes.

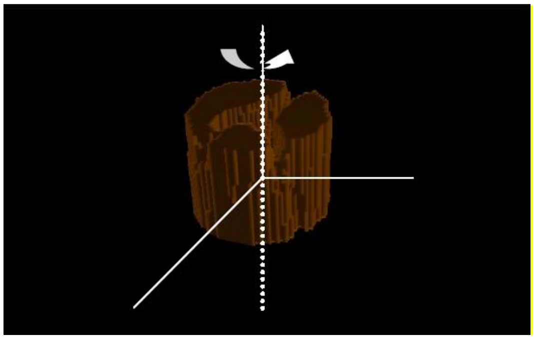


Figure 13. Result showing the rotation around the dotted line defined between the geographical position of the LEDs in the coordinate (0.0, 1.0, 0.0) and the origin of the reference axes.

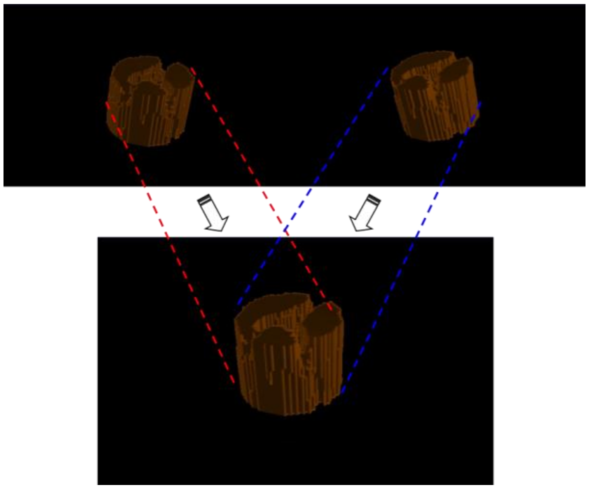


Figure 14. Representation of the rotation described around the stippled axis defined by the coordinate of LED = (1.0, 1.0, 0.0), and passing by the origin of the axes.

Through the *Visualization* class, the three-dimensional samples were examined by the Head Mounted Display in an immersive way. First, *Canvas3D*, responsible for the rendering of three-dimensional images, was maximized to omit the parts related to the main interface in the device, in order to focus only in the region where the sample was showed. Thus, each display of HMD forms an image, which is showing and interpreted by the user's brain with a larger depth effect.

Secondly, such an effect also has allowed performing the analyses of the preferential paths of the water flow into the agricultural soil samples, called as fingering effects, as well as the verification of the percentage of pores in the samples.

As described in the *Non-conventional Scene Manipulation* class, as the cameras are moved with the navigation processes, activated by keyboard interaction or data glove *P5Glove*, the traveled paths can be demarcated; leaving registered the itinerary, under visual and mathematical form.

Furthermore, for each device movement identified, it is established a new position for the camera, that means, given by new coordinates (x, y, z).

Such positions are unique and occupied only one per time. Thus, activated the demarcation process, from any point, the traveled path can be simulated for a certain water flow, i.e., when working with an agricultural sample. When accomplishing a certain movement, the current point occupied by the camera receives a *Shape3D* under the form of a blue sphere, which simulates the presence of a fluid drop occupying the previous position of the camera, leaving a bluish trace through where the camera passed.

In similar way to the simple scene manipulation, such demarcation obeys the laws imposed by the *Non-conventional Collision* class, it means, the traveled path is prevented of passing over the non-porous faces of the agricultural sample, leading the flow of fluids to pass among the related pores, the preferential paths.

The process can be repeated countless times, in way similar to the real situations.

Also, it is possible to make a borders calculation, where the limits of three-dimensional samples are identified in the space, as in the *Attributes Extraction* class, through the *getBounds* use on *Shape3D* instances, combined to a three-dimensional borders detection algorithm called *Polytope*, available in the Bounds package of *Java3D* API.

Such algorithm takes charge of drawing countless plans around of the surfaces of the sample, traveling all its extension in order to delimit exactly its borders. In such a way it allows that the nonporous parts of the samples, including the internal ones, can be identified, allowing the verification of its volume in cm³.

Figure 15 presents the result of the case study based on a tomographic image from a degraded agricultural soil, where the sample is in gray tones, and the water flow is represented with a blue color, demarcating the traveled paths.

In fact, once identified the non-porous part, the remaining portions were recognized based on the emptiness of the sample, which present the color that corresponds to those voxels in which occurred the absence of the photons attenuation. The porous voxel was filled out with a semi-transparent yellow color, seeking a larger prominence close to the sample. With such available data, it is possible to calculate the total volume of the sample (sum of the non-porous parts with its complement) in cm³.

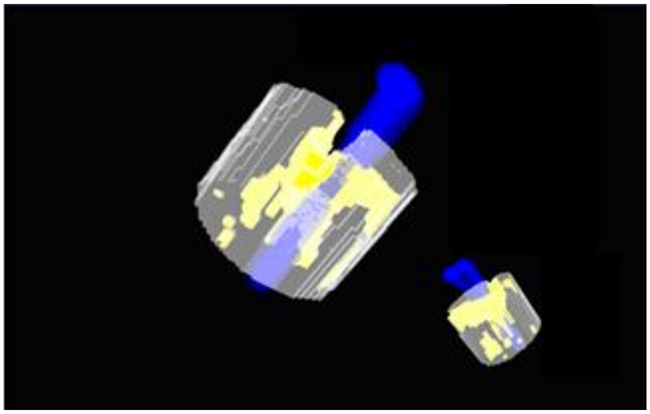


Figure 15. Result of the case study using a degraded soil sample, i.e., with representations of the non-porous soil portion (gray), the emptiness (yellow) and the water flow (blue) in between the soil pores.

Thus, starting from the total volume and the individual volume of the non-porous part, it is possible to calculate exactly the volume represented by the emptiness of the three-dimensional sample.

IV. CONCLUSIONS

This work presented the development of a new method, which took into account the integration of a sensors-based VR environment with a CT for dedicated inspection of agricultural soils. Results have shown both the possibility to access CT digital images from the agricultural soils, and the opportunity to handling three-dimensional manipulation and graphic visualization processes, through computational devices. Besides, such a developed method allowed the addition of the immersion and the user`s interaction with soil samples. Such resources had involved rendering control, illumination, coloring, attributes extraction and physical transformation, besides the integration of non-conventional data input and output devices, such as a *Head-Mounted Display* (video-helmet), and digital gloves.

In addition, it has been also observed that *Java3D* API provided in its group of classes, essential methods for HMD programming. Such development has encapsulated practically the entire stereoscopy programming.

Furthermore, the case study demonstrated the applicability of the method in visualization processes and agricultural soil sample analysis, considering the progresses and facilities when accomplishing non-invasive inspections.

Finally, the integration of CT and the sensors-based VR made possible to measure the volumes of emptiness of the samples, i.e., the pores, and simulate the water flow path for the formation of the preferential fingering. Future works will take into account embedded systems based on the use of *Field Programmable Gate Array* (FPGA), as well as use of the augmented reality concepts.

ACKNOWLEDGMENT

This research was partially supported by the São Paulo Research Foundation (FAPESP, Process No. 17/19350-2), and the Brazilian Corporation for Agricultural Research (Embrapa, Process No. 11.14.09.001.05.06). We thank the institutional support received from the Computer Science Department of the Federal University of São Carlos (UFSCar).

REFERENCES

[1] L. C. Botega and P. E. Cruvinel, “Sensors-Based Virtual Reality Environment for Volumetric CT Analyses of Agricultural Soils Samples”, *Proceedings of the IARIA, ALLSENSORS 2020: The Fifth International Conference on Advances in Sensors, Actuators, Metering and Sensing*, Valencia, Spain, 21-25 November 2020, pp. 27-34.

[2] A. Petrovic, J. Siebert, and P. Rieke, “Soil bulk density analysis in three dimensions by computed tomographic scanning”, Soil Science Society of America Journal, vol. 46, no. 3, pp. 445–450, 1982.

[3] J. M. Hainsworth and L. A. G. Aylmore, “The use of computer-assisted tomography to determine spatial distribution of soil water content”, Australian Journal of Soil Research, no. 21, pp. 435–440, 1983.

[4] S. Crestana, S. Mascarenhas, and R. Pozzi-Mucelli, “Static and dynamic threedimensional studies of water in soil using computed tomographic scanning”, Soil Science, vol. 140, no. 5, pp. 326–332, 1985.

[5] P. E. Cruvinel, R. Cesareo, S. Crestana, and S. Mascarenhas, “X-and gamma-rays computerized minitomograph scanner for soil science”, IEEE Transactions on Instrumentation and Measurement, vol. 39, no. 5, pp. 745–750, 1990.

[6] Á. Macedo et al., “Wood density determination by X and gamma ray tomography”, International Journal of the Biology, Chemistry, Physics and Technology of Wood, vol. 56, pp. 535–540, 2002.

[7] A. Pedrotti et al., “Computed tomography applied to studies of a planosoil” (Original in Portuguese: Tomografia compu-

tadorizada aplicada a estudos de um planossolo). Brazilian Agricultural Research Journal, vol. 38, no. 7, pp. 819–826, Brazil, 2003.

[8] P.E. Cruvinel, M. L. F. Pereira, J. H. Saito, and L.F. Costa, “Performance optimization of tomographic image reconstruction based on DSP processors”, IEEE Transactions on Instrumentation and Measurement , vol. 58, pp. 3295-3304, 2009.

[9] J. M. Beraldo, F. A. Scanavinno Junior, and P. E. Cruvinel, “Application of X-ray computed tomography in the evaluation of soil porosity in soil management systems”, Engenharia Agrícola, vol. 34, no. 6, pp. 1162–1174, 2014.

[10] E. S. B. Ferraz and R. S. Mansell, “Determining water content and bulk density of soil by gamma-ray attenuation methods”, Technical Bulletin, no. 807, IFAS, Florida, pp. 1-51, 1979.

[11] C. F. A. Teixeira, S. O. Moraes, and M. A. Simonete, “Tensiometer, TDR and neutron probe performance in the determination of soil moisture and hydraulic conductivity”, (Original in Portuguese: Desempenho do tensiômetro, TDR e sonda de nêutrons na determinação da umidade e condutividade hidráulica do solo), Brazilian Journal of Soil Science, vol. 29, pp. 161–168, 2005.

[12] M. F. L. Pereira and P. E. Cruvinel, “A model for soil computed tomography based on volumetric reconstruction, Wiener filtering and parallel processing”, Computers and Electronics In Agriculture, vol. 111, pp. 151-163, 2015.

[13] K. Slavakis, G. B. Giannakis, and G. Mateos, “Modeling and Optimization for Big Data Analytics”, IEEE Signal Processing Magazine, pp. 18–31, 2014.

[14] V. Bolón-Canedo, N. Sánchez-Maroño, A. Alonso-Betanzos, “Recent advances and emerging challenges of feature selection in the context of big data”, Knowledge-Based Systems, Elsevier, vol. 86, no.9, pp. 33–45, 2015.

[15] A. Ali, G. A. Shah, M. O. Farooq, and U. Ghani, “Technologies and challenges in developing machine-to-machine applications: A survey”, Journal of Network and Computer Applications, vol. 83, pp. 124–139, 2017.

[16] S. S. Andrews, D. L. Karlen, and C. A. Cambardella, “The Soil Management Assessment Framework: A Quantitative Soil Quality Evaluation Method”, Soil Science Society of America Journal, vol. 68, pp. 1945– 1962, 2004.

[17] A. Kaloxylos et al., “Farm management systems and the future internet era”, Computers and Electronics in Agriculture, vol. 89, pp. 130– 144, 2012.

[18] U. Zimmermann et al., “A non-invasive plant-based probe for continuous monitoring of water stress in real time: a new tool for irrigation scheduling and deeper insight into drought and salinity stress physiology”, Theoretical and Experimental Plant Physiology, vol. 25, no. 1, pp. 2-11, 2013.

[19] J. S. Selker, L. Graff, and T. Steenhuis, “Noninvasive time domain reflectometry moisture measurement probe”, Soil Science Society of America Journal, vol. 57, no. 4, pp. 934-936, 1993.

[20] F. Palacios, M. P. Diago, and J. Tardaguila, “A non-invasive method based on computer vision for grapevine cluster compactness assessment using a mobile sensing platform under field conditions”, Sensors, vol. 19, no. 17, pp. 3799-3818, 2019.

[21] H. Liu, R. Jia, X. Zhou, and L. Fu, “Virtual assembly of man-machine interactive mechanical seed-metering device based on matter-element identification”, Transactions of the Chinese Society of Agricultural Engineering, vol. 32, no. 1, pp. 38-45, 2016.

[22] J. Bouma, “Measuring the conductivity of soil horizons with continuous macropores”, Soil Science Society of America Journal, Madison, vol. 46, pp. 438-441, 1982.

[23] Y. Mualem, “A new model for predicting the hydraulic conductivity of unsaturated porous media”, Water Resources Research, vol.12, pp. 2184-2193, 1976.

[24] M. Kutilek and D. R. Nielsen, Soil Hydrology, Cremlingen-Destedt: Catena Verlag, 1994.

[25] R. A. Bolt, “Put¬that¬there: Voice and gesture at the graphics interface”, in 7th International Conference on Computer Graphics and Interactive Techniques, Washington, USA, pp. 262–270, 1980.

[26] J. Lanier, Visual programming languages, Scientific American, 1984.

[27] L. C. Botega and P. E. Cruvinel, “Development of a Virtual Reality Environment for Agricultural Soil Analysis” (Original in Portuguese: Desenvolvimento de Ambiente de Realidade Virtual para Análise de Solos Agrícolas), in Proceedings of the Workshop of Virtual and Augmented Reality, Itumbiara, Brazil, 2007.

[28] K. Pimentel and K. Teixeira, Virtual reality through the new looking glass, McGraw-Hill, New York, 2nd edition, 1995.

[29] O. Gonzalez et al., “Development and assessment of a tractor driving simulator with immersive virtual reality for training to avoid occupational hazards”, Computers and Electronics in Agriculture, vol. 143, pp. 111-118, 2017.

[30] L. Jacobson, Garage Virtual Reality, SAMS Publication, Indianapolis, 1994.

[31] C. Machover and S. Tice, “Virtual Reality”, IEEE Computer Graphics and Application, vol. 14, no.1, pp. 15-16, 1994.

[32] Sun Microsystems. Java3D Documentation. [Online]. Available from: http://java.sun.com/javase/technologies/desktop/java3d. Accessed December 2020.

[33] C. Kak and M Slaney, “Principles of computerized tomographic imaging,” New york: The Institute of Electrical and Electronics Engineers, Inc., IEEE Press, 1988.

[34] I. Daubechies, “Ten lectures on wavelets”, CBMS-NFS Regional Conference Series in Applied Mathematics, Philadelphia, PA: Society for Insdustrial and Apllied Mathematics (SIAM), vol. 61, 1992.

[35] T. N. E. Greville, “Spline functions, interpolation and numerical quadrature”, Mathematical Methods for Digital Computers, Vol.2, A. Ralston and H.S. Wilf, eds., Wiley, New York, Ch. 8, pp. 156-168, 1967.

[36] S. Chen, L. Xu, and H. Li, “Research on 3D modeling in scene simulation based on Creator and 3dsmax,” in IEEE International Conference, vol. 4, pp. 1736–1740, Mechatronics and Automation, 2005.

[37] E. F. S. Montero and D. J. Zanchet, “Virtual reality and medicine” (Original in Portuguese: Realidade virtual e a medicina), Brazilian Surgical Act, vol. 18, no. 8, pp. 489-490, 2003.

[38] C. Kenner. Essential reality P5glove sumary: Dual mode driver programming: [Online]. Available from: htpps//scratchpad .fandom.com/wiki/P5_Glove:Drivers_etc. Accessed December 2020.

[39] S. C. Biasi and M. Gattass. Use of Quaternions to represent 3-D rotations. (Original in Portuguese: Utilização de quatérnios para representação de rotações em 3-D), Catholic University of Rio de Janeiro, 2002. [Online]. Available from: http://www.tecgraf.puc-rio.br/~mgattass. Accessed December 2020.