CHISel 2.0 User Manual

J.C. Torres, D. Martín, C. Romo, L. López,
P. Cano, A. León

Laboratorio de Realidad Virtual

Universidad de Granada

http://lrv.ugr.es/chis

This document has been edited using Lyx and Latex on Linux.

Abstract

CHISel is a *Cultural Heritage Information System*. It has been designed to manage non Geo-referenced information related to cultural heritage sites and artifacts.

CHISel information is organized as a set of layers referencing to points on the artifact surface. Managing information means editing, rendering, querying, composing and analyzing. If you are currently using Geo-referenced data it would probably be better to use a GIS.

This document describes the main functionality of CHISel version 2.0.

To cite this document, please use: J.C. Torres, D. Martín, C. Romo, L. López, P. Cano, A. León: CHISel 2.0 User Manual. University of Granada. http://lrv.ugr.es/chis (2014).

Forewords

Conventional information systems are frequently used to document Cultural Heritage sites. These systems are adequate for some specific purposes as they can store and retrieve a large amount of heterogeneous documents.

Nevertheless, this is not enough for research and conservation. Researchers need to be able to find out the relationships between data, and very frequently these relationships are spatial relationships that can be established using a 3D representation of the artifact. A new kind of information system is needed to achieve this goal, for which the 3D representation of the object must be the blackboard on which all the data is represented.

This document describes the use of CHISel, a documentation tool that is designed as a GIS-like software for 3D cultural heritage, implementing the concept of a Cultural Heritage Information System (CHIS).

This is an experimental system. It has been designed to evaluate a new concept, and so it has not been programmed as a final product. Nevertheless it has been used for real applications and we hope it can also be useful for you. We have a large list of things we plan to change on it, and so we will be happy to receive feedback from you, as for example which operations do not work as expected or which functions you would like to see included in the system.

We plan to rebuild the system kernel and the model representation, but we will ensure compatibility with the current data format.

The CHISel project has been funded by the Andalusian Science Ministry (Consejería de Innovación Ciencia y Empresa de la Junta de Andalucía) under grant PE09-TIC-5276, and developed by the following research team:

From the Computer Graphics Research Group at Granada University:

- Juan Carlos Torres (project leader)
- Germán Arroyo
- Pedro Cano
- Anibal García
- Alejandro José León
- Luis López

- María Victoria Luzón
- Domingo Martín
- Francisco Joaquín Rodríguez
- Celia Romo
- Francisco Soler

From the Patronato de la Alhambra y el Generalife:

- Elena Correa Gómez
- Francisco Antonio Lamolda
- Ramón Francisco Rubio

From the Conjunto Arqueológico de Itálica:

- Julia Patricia Herce
- Rocío Izquierdo
- Álvaro Jiménez
- José Ramón López
- Juan Bosco Martínez
- Marta Pérez
- Sandra Inmaculada Rodríguez

License and disclaimer

This is an open source software developed as a conceptual test to evaluate a new approach to managing Cultural Heritage information, specially that related to the restoration and conservation processes.

Contents

List of Figures

Chapter 1

Introduction

Managing information related to cultural heritage sites is an important task and much work has been devoted to developing special purpose document management systems. These systems are able to store and retrieve large amounts of documents; however, while this is adequate for some purposes, it is not sufficient for research and conservation work. Researchers need to determine relationships between data, and the most important relationships in cultural heritage information are spatial relationships. A new kind of information system is therefore needed, in which the 3D representation of an object is a blackboard on which all data is represented. We proposed to develop a special kind of Information System applied to Cultural Heritage [Torr12a]. Here we describe the use of our first implementation of the system, for a more technical description of the system please read [Sole12, Torr12].

CHISel is a prototype software of information system that associates information to the surface of a Cultural heritage artifact. It works like a GIS, but it is not a GIS:

- CHISel information is not georeferenced.
- CHISel can manage information linked to any surface.

The following section describes some application examples where CHISel can be used.

1.1 Application examples

This section aims to give a fast overview of the capabilities of the system. It describes previous work that has been carried out using CHISel. It does not cover the whole functionality of the system, but can help you to decide whether CHISel can be useful for you.

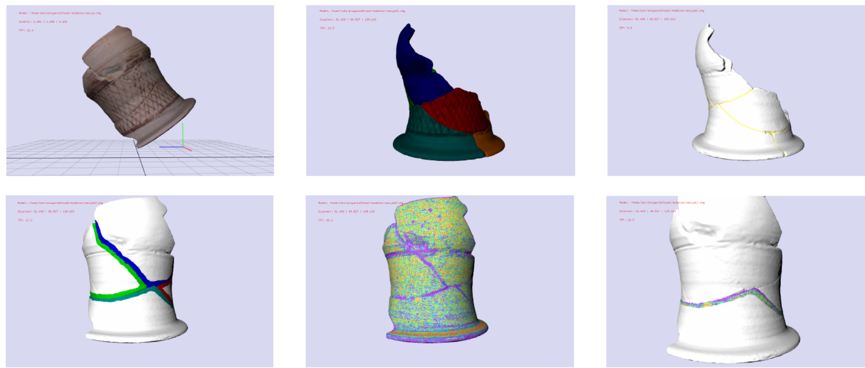


Figure 1.1: Comparing curvature at fragments edge.

Example 1

This example demonstrate the use of the system to compare the curvature value of different fragments of an Iberian vessel. It can be used to determine whether each fragment has suffered a different erosion process. The method applied implies identifying the fragment (figure 1.1top center), and the separation between them (top right). Then a narrow band at the joining part of each fragment is computed (left bottom). Curvature is computed from geometric information, and finally the curvature at the border of each fragment is measured. A detailed description can be found in [Torr12].

Example 2

Restoration processes modify roughness and curvature. Figure 1.2 shows the curvature value for the two wings of and angel sculpture. The right wing has been restored. A detailed description can be found in [Lope13].

Example 3

A similar process has been carried out with the sculptures of the Lions at the “Patio de los Leones” in the Alhambra. Here the roughness was analyzed in three different areas comparing them with the closest area of the sculpture (green region on figure 1.3 left). This band was computed from a distance field shown on the right image. A detailed description of this example can be found in [Torr13].

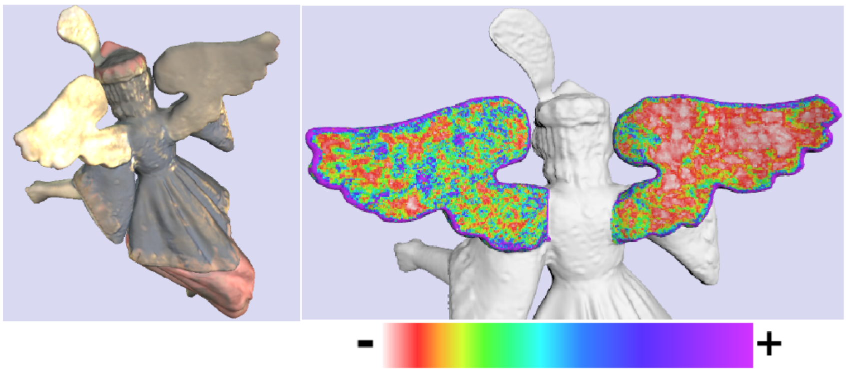


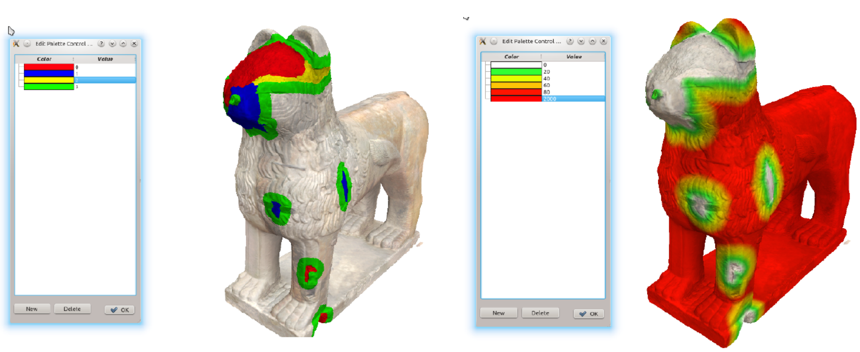
Figure 1.2: Analyzing restoration of the right wing

Example 4

CHISel can also be used to log restoration processes, keeping track of operations performed. Figure 1.4 shows a log layer of a restoration process. This information can be queried and analyzed in a very flexible way [Torr13].

Example 5

Figure 1.5 shows the use of curvature information to reveal damaged areas on a mosaic at the Amphitheater of Italica [Torr13a].



	Average	Maximum	Standard deviation
Loss of Material	0.139	1.496	0.248
Added Material	0.122	1.248	0.229
Erosion	0.157	1.083	0.284
Band	0.140	1.455	0.266

Figure 1.3: Comparing roughness at different areas.

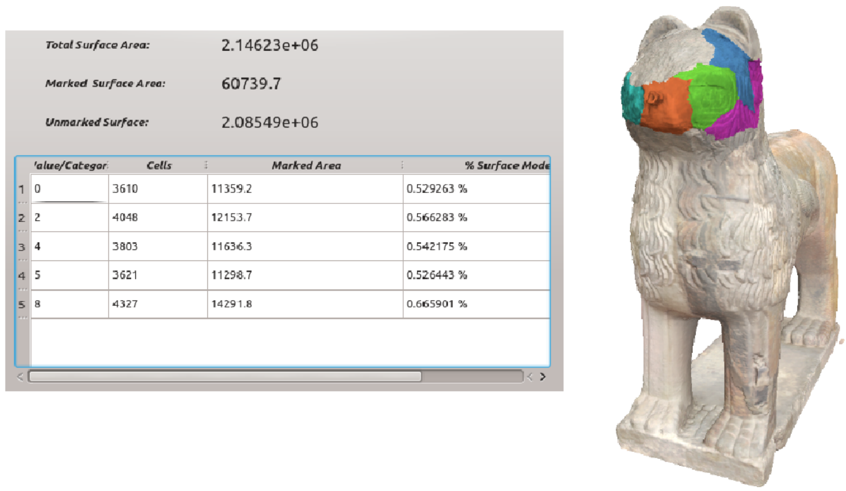


Figure 1.4: Restoration log.

Example 6

Orientation information can be used to generate segmentation layers. Image 1.6 shows the result of the segmentation of a room at the Amphitheater of Italica [Torr13a].

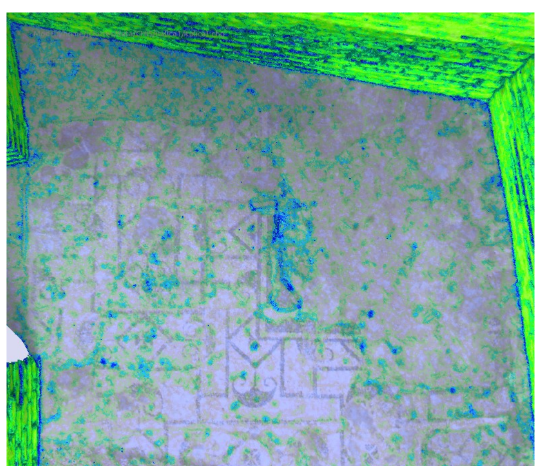


Figure 1.5: Revealing damaged areas.

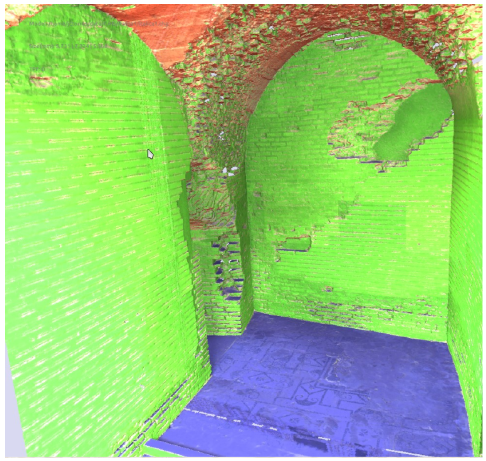


Figure 1.6: Automatic segmentation.

1.2 Installation

For most systems you can obtain an executable file for CHISel. Just copy it onto your system and run it. If you have also got a CHISel model you can test it directly. Otherwise you will need to start importing your 3D models. It is recommended to

- create a Models directory inside the Installation directory used for CHISel
- create a subdirectory for every new model inside the Models directory
- use a different name for every new model
- do a backup of the whole system (including CHISel installation).

Chapter 2

Loading models

The system has been designed to create representations of artifacts from digital models generated using a laser scanner. It can read a 3D model stored using the PLY format. PLY is an open format that was specifically designed to store three dimensional data of 3D scanned objects. A PLY model stores a geometric description of the surface of an artifact, which is represented as a triangular mesh.

The application can automatically Import the surface description of an artifact from a PLY file, computing its cell decomposition and its topological relationships. The user must specify the maximum desired resolution, in other words the minimum size of the cells. The prototype can typically manage representations of objects with a cell size of 1 mm for an object whose dimensions are more than some meters. This seems to be enough for most practical applications. Once an artifact is imported from a PLY file, the application uses its own object representation which is saved to disk with the specific application file format. Figure 2.1 shows an overview of the application.

2.1 Importing a model

CHISel can load binary PLY models. Binary PLY models can be generated from many different 3D formats using MeshLab. To convert the format using MeshLab, import the model using

File > Import Mesh

and export it using

File > Export Mesh

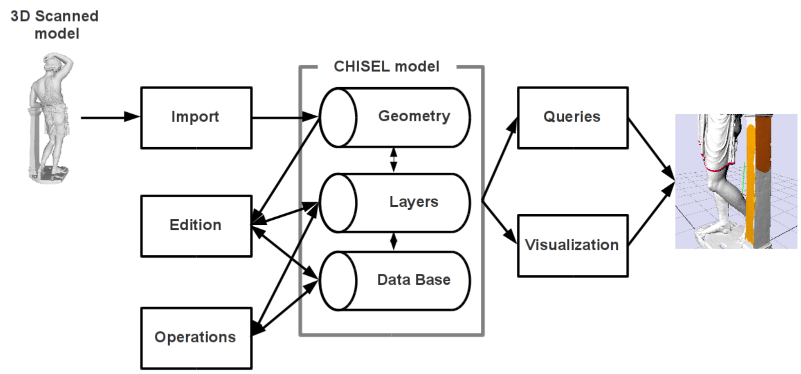


Figure 2.1: Overview of the application.

Check binary format and the color checkbox, see figure 2.2. Color checkbox will not be available if the model has no color information. In this case the model imported in CHISel will not have color texture.

Once the model is exported you can close MeshLab and open CHISel. To run CHISel just double click on the CHISel icon, or open a shell and run the program from it.

On CHISel menu select the option

File > Import Model

This will open a file selection dialog. Once you select the file, an import dialog is displayed (see figure 2.3).

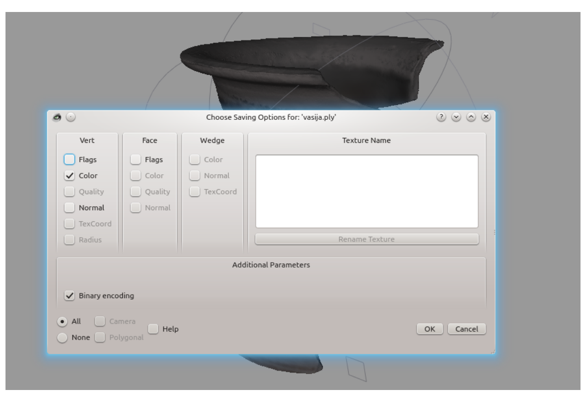


Figure 2.2: Exporting a model as binary ply file using MeshLab.

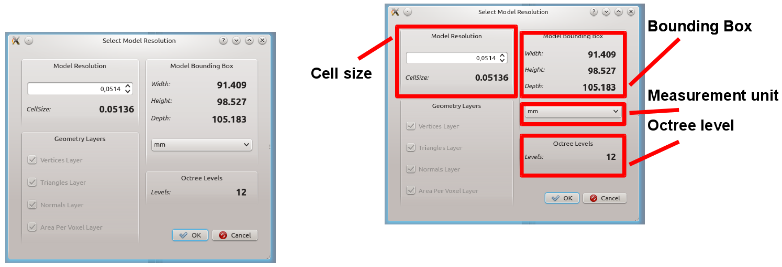


Figure 2.3: Importing a PLY file in CHISel. Right: Sections on the import widget.

This dialog is used for several purposes:

- To inform you about the model size (formally its bounding box, the minimum orthogonal box containing the model) and its bounding box, see figure 2.3.
- It allows you to specify the measurement Unit used on your data (see measurement unit area in figure 2.3).
- It allows you to specify the maximum resolution you can use on the layers that will be defined in the future (see area measurement unit in figure 2.3).
- It informs you about the depth of the tree structure that will be used to index your model (see octree level area in figure 2.3).

Measurement units are not stored in PLY files, they just contain numbers. So you must inform the system if your coordinates are centimeters, millimeters or meters. You can obtain this information from your capturing device, or calculate it from the information provided for the bounding box. The default value is millimeters.

Remark 1. The current version uses only the metric system.

Once you have specified the measurement unit, you must select the maximum resolution level. This will determine the depth of the indexing tree.

You will be able to create information layers with lower resolution (larger cell sizes), but not with higher ones. Select the resolution by entering any number (or increasing or decreasing it) on the model resolution input box. The system will display the level of the index tree (octree level), and the real cell size that will be generated using this octree.

Remark 2. The memory and time required to import the model depend on the octree level. Do not use more resolution than needed.

Press the ok button to import the model. Importing can last from a few minutes to several hours depending on the octree level and the number of triangles in the model. When the model is imported you will see an image similar to figure 2.4.

Once the model is created it is recommended to save it as a CHISel model. To do so, use the option

File > Save model as...

Please use a new directory to store the CHISel model. Do not use the same directory to store two different models or to store any other information. The default file name is chisel. Please use a different name for every model.

Remark 3. It is strongly recommended to close the application when the model is saved, and reopen it to load the new model using

File > Load Model

2.2 CHISEL model representation

Once an artifact is imported from a PLY file, the application uses its own object representation which is saved to disk with the specific application file format.

CHISel models are composed of four main components: a 3D model, a spatial index, a set of information layers and a database (see Figure 2.5).

The 3D model holds the geometry of the site, stored as a triangle mesh. For every vertex the model contains its position, normal and color.

The spatial index divides the 3D model into a set of surface patches and assigns a unique identifier to each one. These patches, called cells, are created

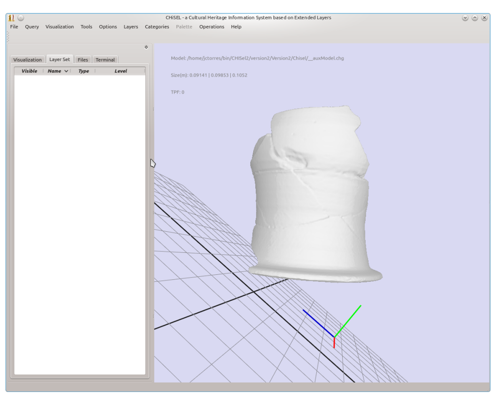


Figure 2.4: Screen shot of the application when a new model has been imported

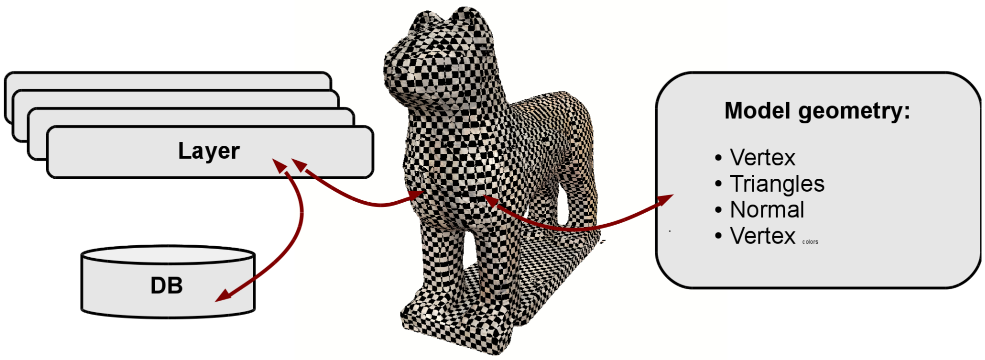


Figure 2.5: CHISel data model

by intersecting the 3D model with a regular grid [Torr12]. Every patch has a maximum size that is defined by the user. The cells have a unique identifier assigned (Figure 2.5). Layers are stored as a sequence of values. It should be noted that although all the cells have the same size, the area of the surface within each cell is different. That is, cell size determines the maximum area of the surface that will have a value assigned. All computations are carried out taking into account the area covered by the object surface in the cell.

Cell identifiers are assigned automatically by the system. This allows data to be assigned to the surface and to create correspondences between different information layers.

This representation allows raster layers to be managed flexibly, in a similar way to a GIS raster map.

Information layers relate attribute values to the patches of the object surface. A layer contains values assigned to patches. The associated values can be numbers or records from a relational database.

The relational database contains complex information (text, images or files) that is associated to the model through the layers and the spatial index.

This structure allows us to establish a bidirectional relationship between the database information and the model geometry.

The representation of the CHISel model contains three connected components: geometry, data layers and database. When the model is created from an external file it contains only geometric information. The user can create information layers in several ways: interactively, from geometric information or from previously created layers.

Raster layers can hold null values, implemented as a bits array, with the same size as the layer array, indicating whether the stored value is valid.

2.3 Files structure

CHISel models are stored as a set of files plus a SQLite relational database. The current version stores a primary file with extension "chg" that contains the configuration of the model and the set of files used.

Every model is stored in a different directory, whose name is selected by the user and is the same used for the primary file.

Remark 4. File names are assigned automatically by the system, they are the layer name with extension (.lay). The files can be is stored using binary or ASCII format. Deleting or renaming them will corrupt the representation.

If you want to make a backup of the model you can zip the whole model directory.

2.4 Topological information

Cell connectivity information is computed automatically and stored using a special layer that contains the identifiers of neighboring cells for each cell.

Topology is used to compute distance fields and to compute statistic information on the neighboring area.

Chapter 3

Layers

A layer is the representation of a property related to the object surface that associates values to surface patches. Surface patches for which the property value is unknown, or is not defined, have a null value for the property.

The property can be a qualitative or quantitative data. Quantitative data are stored directly as numbers (real or integer) in the layer array, while qualitative data are stored as database records. A qualitative datum is a complex attribute represented as a register. In this situation, the layer array contains the primary key for the record associated with the cell. The application can deal with text, dates, numbers, images, film and any other document as a database record field.

The database contains a table for each qualitative layer, whose fields are defined by the user when the layer is created. Fields can be: numbers, text strings, dates or external files (images, pdf files, spreadsheets or whatever). The system includes a numeric identifier as a key for each table record. This key is stored in the layer array.

Remark 5. The application copies the external files into the model directory.

3.1 Types of layers

Layers can be classified according to the information they can associate to each cell:

- Raster Layer: associate no more than one value per cell. Some cells can have no value (or null value).

- Vector layer: can associate more than one value to a cell. They are equivalent to a points vector map.

The information they associate to the cell can be a number or a database record. So we consider

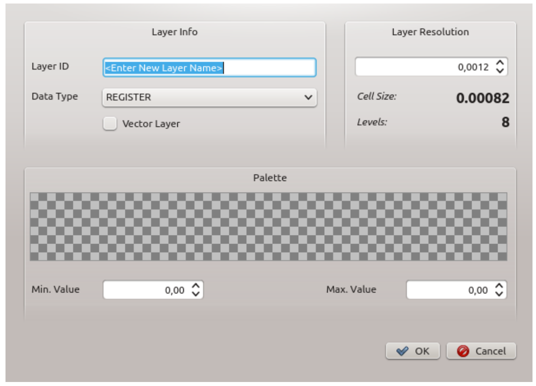


Figure 3.1: Creating a layer

- Quantitative layers: cell values are numbers. The representation for the number (from unsigned integer to double precision) is decided when creating the layer.
- Qualitative layers: cell values are database records. A data base table is defined when the layer is created.

Some layers are created as a result of operations performed on existing layers. This process will be discussed in chapter 6. The following sections focus on the definition of new layers by the user.

Creating quantitative raster layers

To create a new raster layer use the menu options

Layers > New Layer

The dialog shown in figure 3.1 will appear in a pop up window. You must indicate the name for the new layer, entering the "Layer ID" field, the number representation, using the "Data type" selector, and the layer resolution.

- Layer ID can be any string. It must be meaningful to you. This is the name you will see on the interface, it is not used as a file name for the representation.
- Data Type can be an integer (unsigned discrete or signed discrete) or float (single or double precision). The register option (shown by default) is used for qualitative layers and is explained in the next section).
- Layer resolution. This is the cell size for this layer, it is limited by the model resolution.

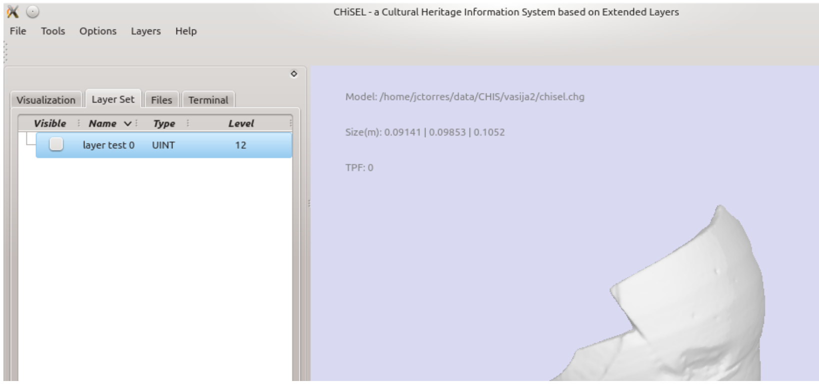


Figure 3.2: New layer at the layer set

- Value range. These values are used to define the interval of valid values. This can be changed by modifying the color palette.

Additional control on this widget allows us to create vector layers (explained in section **??**) and setting the initial color palette for the layer (this will be explained in section 3.3). At this moment we do not need to edit them.

Push the "ok" button and the new layer will be created. You will see it on the layer set tab on the left of the application (see figure 3.2).

Creating qualitative raster layers

To create a qualitative raster layer just set "Data Type" as "Register". When you push the ok button a new dialog will allow you to enter the database table structure (see Figure 3.3).

The structure is edited using the "Add New Field" and "Remove Field" buttons. For each new field you must enter

- its name.
- its type: this can be an integer, a float number (DOUBLE), a text (VARCHAR), a date or a file.

For text file it is necessary to specify its maximum length (Size).

Push the ok button when done. The layer will appear in the layer set area.

This structure can be edited at any moment using the option

Layers > Modify Table Layer

Figure 3.3: Defining the table when creating a register layer

Creating vector layers

For some purposes it is necessary to associate more than one value to some cells. For instance, if we are recording artifacts found at an archaeological site, we must be able to store information about more than one artifact at the same location. A convenient way to take this into account is to incorporate vector layers.

CHISel vector layers hold quantitative or qualitative information

To create a vector layer just check the vector radio button on the layer creation widget.

3.2 Layer edition

The usual way to introduce information into the system is to create information layers. Layers can be created interactively by describing the structure of the associated database table and editing the layer values on the 3D model. The editing process is performed as an interactive "paint-like" operation: in other words, the user selects the areas of the artifact with a specified value by simply "painting" them with a brush.

Take into account that we are adding semantic information to the model, not just colors. Colors are simply a mechanism to visualize the semantic information, as on a thematic map. Section 3.3 explains how to define and manage colors.

To edit a layer select a layer in the layer set widget, the layer row will be displayed on a blue background (see figure 3.4). And then use the option:

Layers > Edit Layer

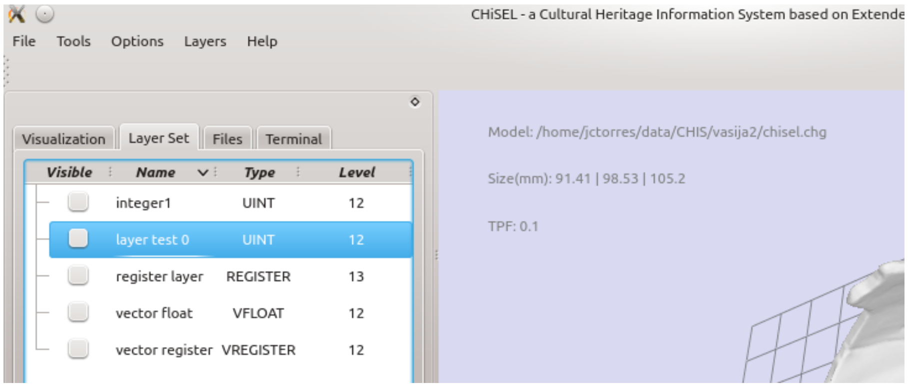


Figure 3.4: Selecting a layer for edition. Layer "layer test 0" is selected.

while the layer is selected.

Remark 6. If the selected layer is of type register it is necessary to define the database records previously.

A pop up window will appear showing a value selector widget. This widget contains a value selector and a box displaying the color assigned to this value. Enter the value you want to assign and transfer it to the model that must hold this value (see figure 3.5).

To assign the actual value press the Control key and click on the area you want to modify using the mouse left button. The cursor is displayed as a circumference. This circle specifies the area to which the actual value will

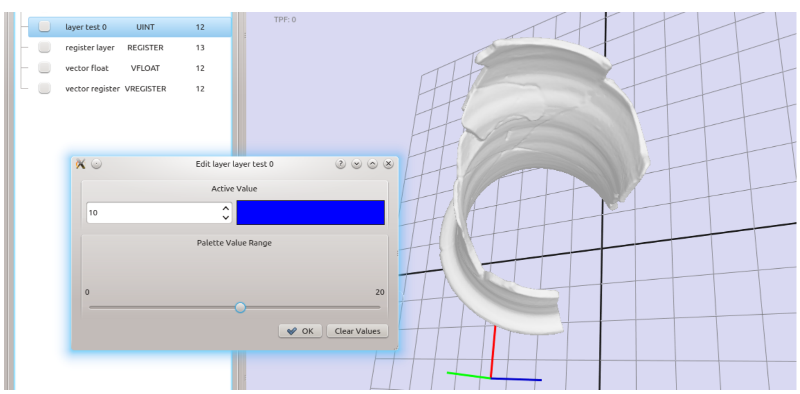


Figure 3.5: Editing a layer. Value 10 is selected to be assigned.

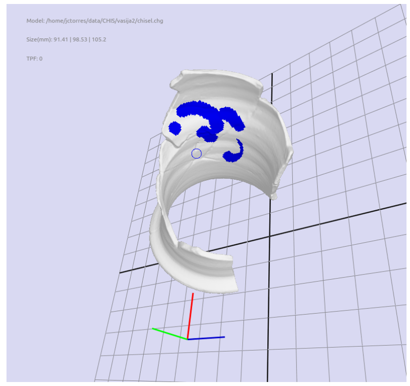


Figure 3.6: Editing a layer. The blue circumference marks the area to be assigned .

be assigned. You can modify the circle size by using the mouse wheel while pressing the Control key.

To change the value to be assigned to a new area just modify the selected value at the widget.

At any moment you can erase values using the “clear values” button on the selector widget.

It is possible to move the camera at any moment using the mouse without pressing the control key (for a description of the camera control see chapter 5).

Press the ok button to finish the edition of this layer.

Remark 7. Only values associated to the layer can be assigned to cells. For a qualitative layer you need to create the corresponding database records (see section 3.2); for quantitative layers you must extend the valid values interval by modifying the color palette (for information on color palette see section 3.3).

The layer will not be displayed when you exit the edition tool. To display it, check the layer check-box at the layer set widget.

Adding categories to qualitative layers

Select a register (qualitative) layer, and then use

Categories> Create Category

This will display a new pop up window (see figure 3.7). All fields, but the Category ID will be empty. Category ID is the table key for this layer, and will be generated automatically as consecutive numeric values. Anyway, you can modify it to assign it a special meaning.

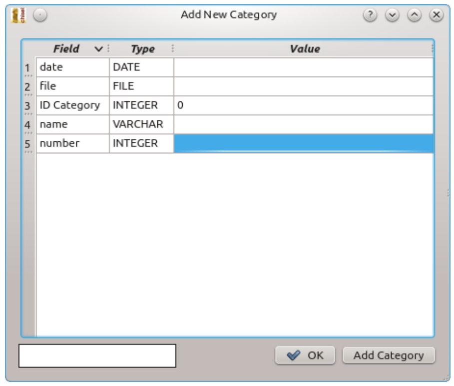


Figure 3.7: Adding a category to a qualitative layer .

You can also assign a color entry for this category in the layer palette by selecting it from the color box at the bottom (initially it will be white). When all the data for this record are correct press *Add Category*. When there are no more categories to enter press *ok*, and the program will ask if you want to add the current record. Say no if you have just added it using *Add Category*.

Data fields are processed as an 8 digit number containing year, month and day as consecutive figures. Always use the same format for comparing dates.

External files are selected from the files that have been imported onto the file tab. To import a file just select this tab, press the right button and select import on the pop-up menu.

There is a folder on the file tab for each qualitative layer and another one for vector layers. Import the file into the corresponding layer folder.

It is also possible to edit and delete categories using

Categories> Edit Category
Categories> Delete Category

If the categories have been used, deleting them will cause a loss of the information on the layer.

Editing vector layers

Vector layers are edited following the same process described for raster layers. The only difference is that you store a new value every time you click on the model. Values are associated to cells, so you cannot change the size of the editing tool. They are rendered at cell level, so you can see only one value even when there are more than one associated to it. Anyway, you will be able to retrieve this information and perform queries on it.

Files that will be referenced in vector layers must be imported into the vector register directory.

Layers management

Layers can be deleted, copied or renamed. To rename a layer double click on its name in the layer set widget. To delete and copy them you have two menu options:

Layers > Delete Layer
Layers > Copy Layer

Remark 8. Deleting a layer does not remove the associated files.

3.3 Layers rendering

Layers are displayed as color information when the model is rendered. For this purpose every layer has an associated color palette that specifies the color to be assigned to every property value. The user can choose which layer must be rendered and whether the model texture is used. The color of every surface pixel is computed by blending the surface texture (if it is selected) and the color assigned to the properties of the cell containing the pixel.

To render the model using its own texture (otherwise it will be rendered using a white color), use

Visualization > Use Model Color

The options menu also contains entries to control the window background and to switch off the display of axes and the background grid

Visualization> Background Color
Visualization> Shows Grid
Visualization> Shows Axis

To render a layer you must check its visible check box in the layer set tab.

Visible layers appear on the visualization tab in the same order they will be rendered. The order depends on the selection time on the layer set tab. If you want to change the order go to the layer set tab, unselect the layers and select then again in the appropriate order.

By clicking on a layer name you can perform two different operations on the visualization tab:

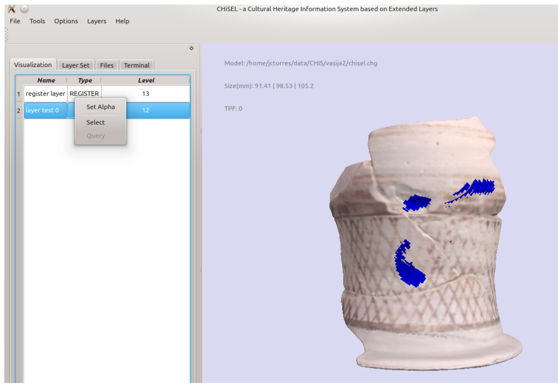


Figure 3.8: Rendering a layer. The pop-up menu allows us to modify the layer transparency.

- setting its opacity. By default, layers are not transparent (its opacity, “alpha” value, is 1). Alpha can have an assigned value between 0 (totally transparent) and one opaque (completely opaque).
- selecting the values that will be shown. This can be done by using the select option on the pop up menu, and entering the desired values. For qualitative layers this can be also done using a SQL query.

Defining the palette

The color palette assigns a color to each property value. Every layer has its own color palette. To manage it you must select the layer in the layer set tab (it will be shown on blue background). Palettes can be saved and loaded using

Palette > Load
Palette > Save

Both options will allow you to select the external file used.

This is a convenient mechanism to allow different layers, even for different models, to use the same colors.

Palettes can be edited using two different methods: as a discrete set of control points or as continuous intervals. Both methods can be used for all types of layers. Independently of how the palette has been defined it can be modified using any method. In fact there is a unique internal representation of the color palette that uses a set of pairs (value,color), fixing the color used

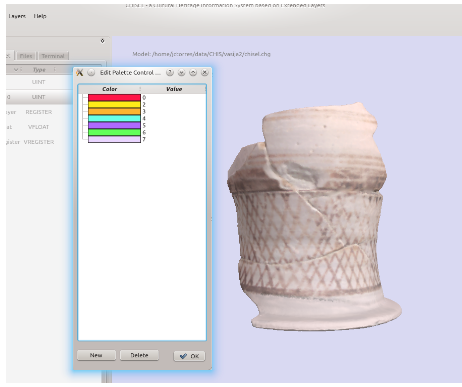


Figure 3.9: Editing a color palette as a set of control points.

for the specified value. For values not included in the list, the color assigned to the two nearest values is interpolated.

To define the palette from a set of control points use

Palette > Edit Ctrl Points

A new pop up will appear displaying a list of layer values to which a color has been assigned (see figure 3.9)

Here you can add and delete pairs (value, color) or change the color assigned to a value by double clicking using the left mouse button on the color (figure 3.10). This will open a color selection widget. Note that you can set a transparent value to a color, which is independent of the general transparency assigned to the layer.

Remark 9. Some pop up widgets block the main window. Be careful: if you select the main window it may appear over the pop up. In this case minimize the main window and close the pop up using the ok button.

For a quantitative layer it could be better to define the palette by editing the intervals, as in this way you can see the interpolation that will be carried on. To do this use

Palette > Edit Ctrl Points

The new pop up widget that appears displays the color assigned to all intermediate values (see figure 3.11).

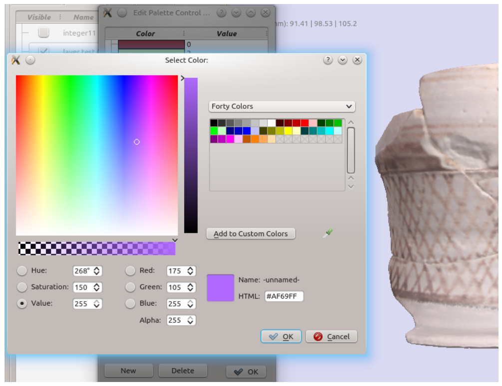


Figure 3.10: Selecting the color.

Control points (pairs value-color) are displayed as small circles. Their position on the graphic represent the property value (for the x-axis) and the opacity (for the y-axis). To edit the palette click on a control point and move it. You can double click at any point and insert a new control point. Double click on a control point to modify the color assigned. Colors are edited in the same way as "Edit Control Points".

Remark 10. Every time a layer is created, a standard palette is assigned to it. Usually you will need to modify it. A convenient way to do this is to assign a previously saved color palette.

3.4 Data base

Chisel data base is an SQLite database, that is stored as a file in the model directory. Every time you create a new qualitative layer a new database table

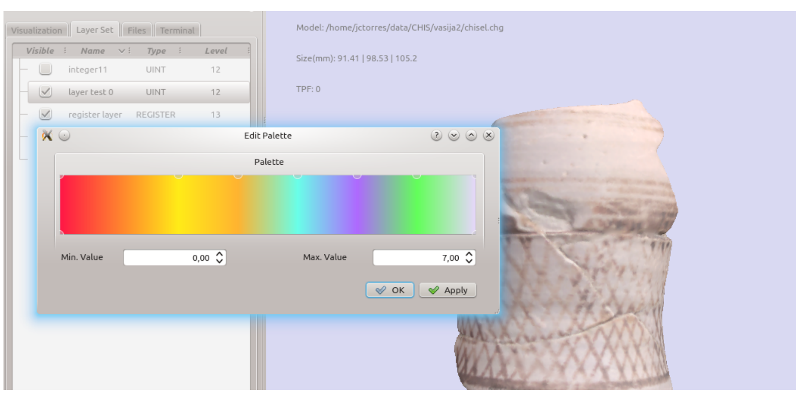


Figure 3.11: Editing the palette displaying the intervals.

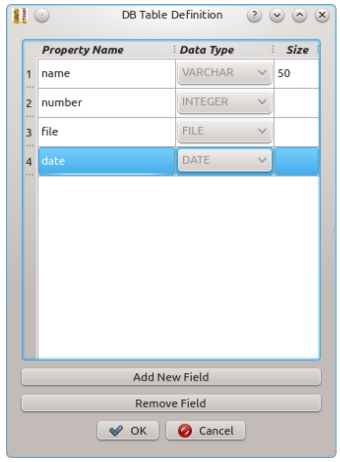


Figure 3.12: Editing the Data Base structure

is created. The fields in the table can be edited at any moment (see section 3.2).

At any time you can also modify the table structure, adding new fields or deleting them. To do so select the layer and use

Layers > Modify Table Layer

The edition pop up window is shown in Figure 3.12.

You can add and remove fields. Removing a field implies losing the information stored in it.

3.5 Layers conversions

It is possible to perform operations on layers (this will be discussed in sections 6 and 7). For matters of efficiency and simplicity, most operations are defined only for numeric layers. So the system includes functions for converting qualitative layers to quantitative ones, and vice versa. To create an scalar layer from a register one use the options:

Operations > Conversion > Register to Scalar

This will open the conversion widget on the terminal tab (see Figure 3.13). Here you must specify the name of the register layer, and select the numeric field that will be used to generate the new quantitative layer. The new layer will have this value as category.

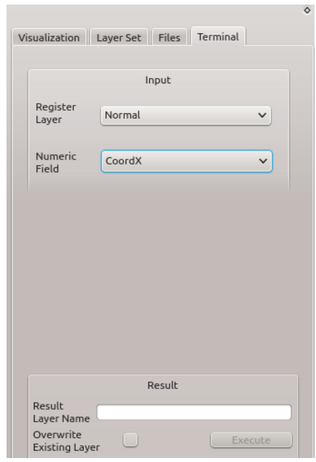


Figure 3.13: Converting a register layer to a quantitative one

The inverse operation is performed using:

Operations > Conversion >Scalar to Register

In this case you can either create a new field in the table for the output layer or modify the values for an existing layer. The output layer must be an existing qualitative layer.

Figure 3.14 shows the widget for this operation.

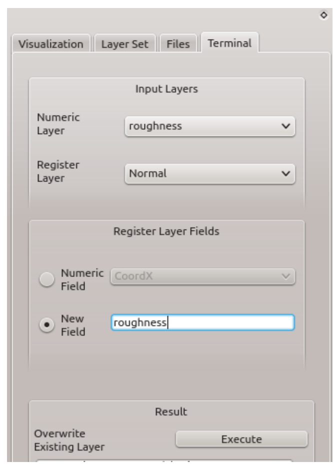


Figure 3.14: Loading a quantitative layer to a field in a qualitative layer

Chapter 4

Queries and reports

Users can obtain the property value assigned to any surface point, in any given layer, by simply clicking on the point. The input information is a surface point, and the output information is the associated record on the database, or the quantitative value associated to the cell containing the point on the selected layer. This kind of query operation is similar to that which is available on most labeled 3D models.

CHISel provides different kinds of queries to be performed:

- Interactive query at a point. The user pick at a point and obtains the information associated to it for the selected layer.
- SQL queries, expressed as a SQL selection on the associated table, and whose result is shown as a 3D visualization of the set of points satisfying the query condition. In this case, the input is a condition and the output is a subset of the surface. Using layer operations it is also possible to perform a condition based query on a quantitative layer (see section **??**).
- Query by value. Display only the areas of the model that have specified values (see section 3.3).

It is also possible to generate reports and to obtain statistical information.

The following section explains the process to perform queries and generate reports.

4.1 Query at a point

This query retrieves the value of a layer at a location defined by the user, which is specified in an interactive way, just by pointing on the model surface using the left mouse button while pressing the control key. To access this operation select the layer and use the commands:

Query > Get Value

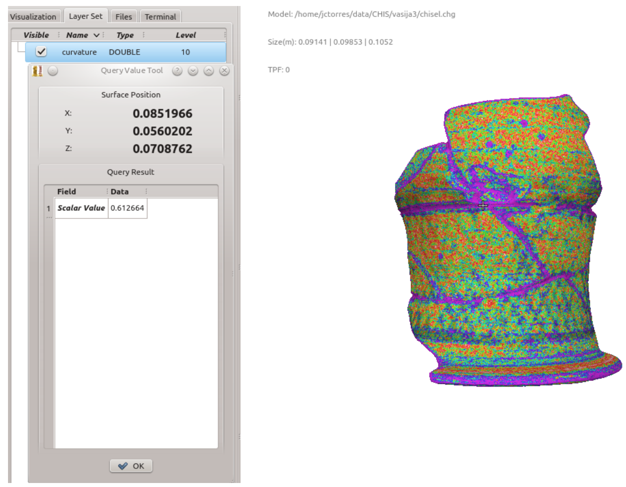


Figure 4.1: Example point query

Results are shown in a po-pup window. Figure 4.1 shows the result of a point query. In the case of a qualitative layer the query shows the value for all the layer fields.

To exit the tool press the *ok* button

4.2 SQL Query

Performing an SQL query on the database generates a sequence of records. CHISel generates a new layer with this information. To access to this operation select the layer and use the commands:

Query > Database Query

Figure shows the corresponding widget.

The *where* clause uses standard SQL syntax.

4.3 Layer area statistic

One of the advantages of using an information system is the ability to generate quantitative information. Statistical functions of any numeric layer can be easily evaluated on the whole model or on a portion of it, computing the value of the function cell by cell, by weighting the result with the cell area.

CHISel implements per category surface area measurement, and multivariate statistics. Surface area measurement reports the area covered by each category of the selected layer.

To access this operation select the layer and use the commands:

Query > Layers Area Stats

Figure 4.3 shows the result of the computation of layer area statistics on the restoration log layer from Figure 4.2 middle.

Results are measured using the model units.

4.4 Multivariate statistics

Multivariate statistics computes a statistical function of a layer for each portion of the model defined by the values of a base layer. CHISel implements average, maximum, minimum, deviation and standard deviation, as statistics operators.

To access this operation use the commands:

Operations > Statistic > Multivariate Statistics

This widget allows us to specify:

- The base layer. This is the layer that defines the group for which the property will be computed. It can be quantitative or qualitative. For qualitative layers you must select the field used to define the groups. If quantitative layers are used they must be of integer type. For float layers you can previously convert them to integer type using map algebra.

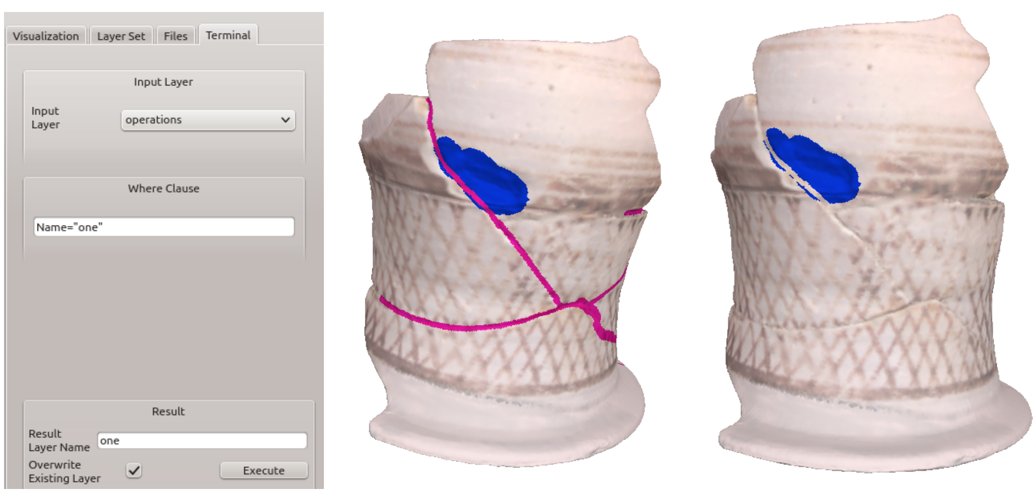


Figure 4.2: Example SQL query. Left widget, middle base layer, right resulting layer.

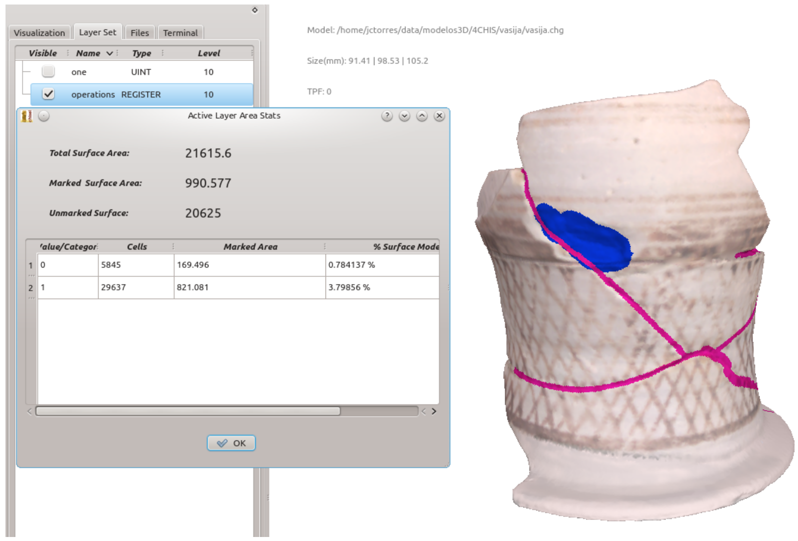


Figure 4.3: Layer area statistic

- The data layer. This is the numeric layer that contains the attribute that will be computed.
- The method. This is the statistic variable that will be computed. You can select all the methods you like at the same time.

As an example, Figure 4.5 shows the computation of all statistic variables for the roughness layer for the different sections of the Vessel defined by the

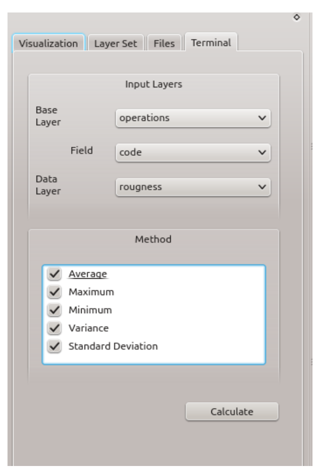


Figure 4.4: Multivariate statistic widget

segmentation shown in the middle of Figure 4.2.

4.5 Legend

Most operations that can be performed generate a new layer containing a result which can be rendered on the 3D model. This is convenient for showing distribution values, but sometimes quantitative results are needed. This figure usually needs a legend. The only way to do this with the current version is to edit the palette or edit layer operation, and capture the pop-up window.

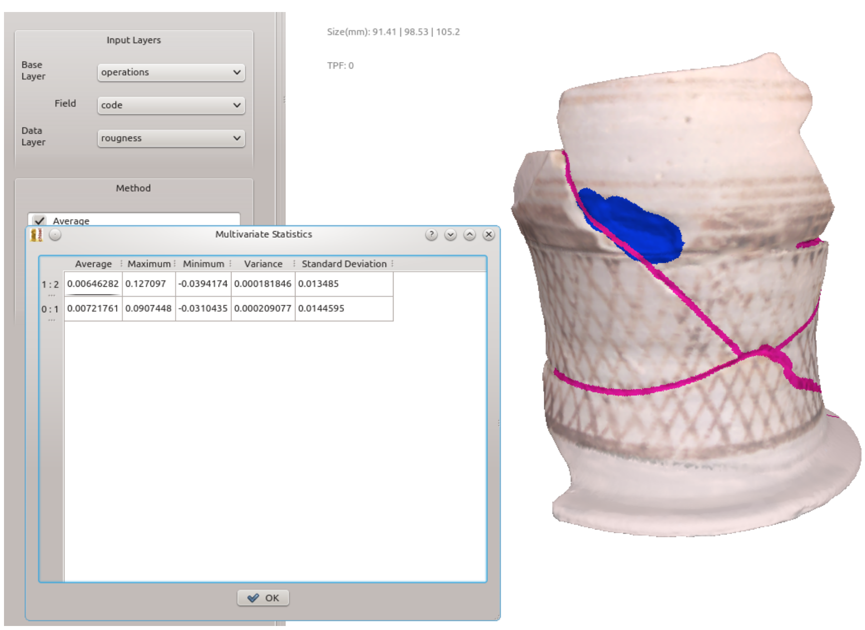


Figure 4.5: Multivariate statistic example

Chapter 5

Rendering the models

Layers are rendered as color applied to the geometric model. The user assigns a color table to every layer, and can select the ordered sequence of layers that will be rendered at any moment as well as whether the original model texture will be visible under the layers color (see section 3.3).

This chapter describes how to control the camera for the 3D rendering widget.

5.1 Camera control

You can rotate the camera around the object, pan the image and scale it.

To rotate the camera move the mouse while pressing the left mouse button.

Moving the mouse while pressing the center button will produce a pan effect.

The middle button wheel can be used to zoom in and out from the object

5.2 Screen shot

There is a snapshot operation that can be located in

Visualization > Capture Image

Figure 5.1 shows the image generated using this method.

You can also use the screen-shot capture of your operating system. This will allow you to capture the legend window also.

5.3 Lighting

You can move the light by moving the mouse while pressing the left button and the shift key at the same time.

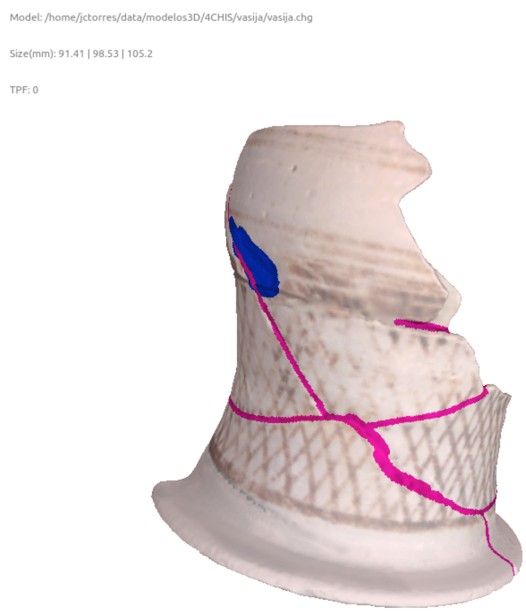


Figure 5.1: Image capture by the system

Chapter 6

Geometric layers computation

The geometry of the model gives us information about its structure, texture and orientation. This information can be useful when analyzing the model.

Data layers and geometry are two different components, but for many problems geometric information is the most relevant one. It is therefore important that geometric information can be used to compute properties, not only as visualization support for layers. That is to say that the analysis process can also be carried out involving geometric information.

6.1 Geometric information in CHISEL

In order to be able to manage geometric information, CHISel includes functions for generating layers containing geometric information about 3D geometric models. Geometric layers are stored in the same way as user defined layers (usually quantitative ones). So when a geometric layer is created it can be managed in the same way as user defined layers. Presently the CHISel prototype can compute the following geometric layers:

- **Curvature.** Computes a layer containing the mean curvature of the surface at the cell.
- **Roughness.** Computes a layer containing the roughness of the surface at the cell.
- **Normal.** Computes layers containing the normal vector component at the cell.
- **Color.** Computes layers containing the rgb color component of the model texture at the cell.
- **Orientation.** Computes a layer containing the angle of the normal to surface at the cell with a given input vector.

These layers must be computed by the user. The following sections explain the creation process.

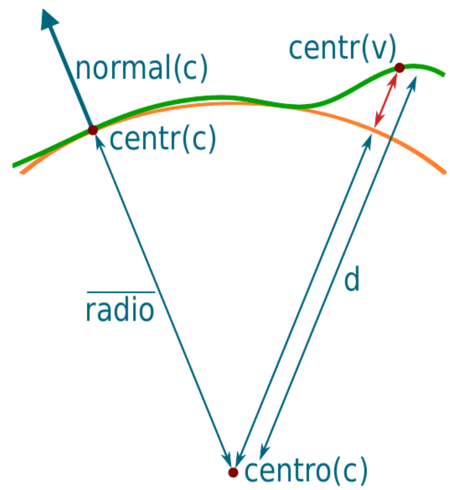


Figure 6.1: Computing curvature

6.2 Curvature

This function generates a layer by assigning a curvature value to each cell. The curvature is defined as the inverse of the tangent sphere to the surface at the cell. The sphere radius is computed as the distance from the surface to the center of the tangent sphere, and the sphere center is computed as the intersection point of the normal at the cell with the normal at neighboring cells. At present, only the main curvature, K, is computed, while it could be possible to compute analytically principal curvatures, K_1 and K_2, as the inverse of the radius of the largest and smallest tangent circles. Figure 6.1 describes the computation process. The normal at the cell and the normal at a neighboring cell are intersected, which gives the center of the tangent sphere to these two cells, which is used to compute the radius. This process is performed using all neighboring cells, and the results are averaged using the surface covered by the object at the cells as weight. A more detailed technical description of the computation of the curvature can be found in [Lope13] (in Spanish).

To compute the curvature map use:

Operations > Geometry > Curvature

This will open the operation dialog on the Terminal tab, see Figure 6.2. This tab includes three inputs:

- Maximum Distance: That is the distance to the cells that will be used to compute the curvature. This distance is measured using the unit

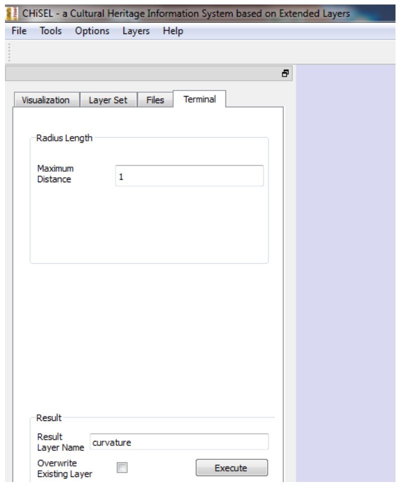


Figure 6.2: Curvature Tab

length used for the model. Note that increasing this distance increases the computation time exponentially.

- Result layer Name: This is the name of the new layer that will be created. It can be any string containing character, numbers and some special characters (including spaces).

- Overwrite Existing Layer: This box must be checked to activate the Execute button when the output layer exits. In this case it will be rewritten.

Pressing the Execute button will create a new layer. The layer resolution will be the working resolution, which by default is the model resolution.

Once the layer is created, the new layer appears on the layer tab, but it is not displayed automatically. To display it, open the layer tab and check the new layer.

Curvature layers are created using a default palette that is shown in Figure 6.3. This palette can be edited. Figure 6.4 shows the curvature layer computed for the vessel model using value 4 for the distance parameter.

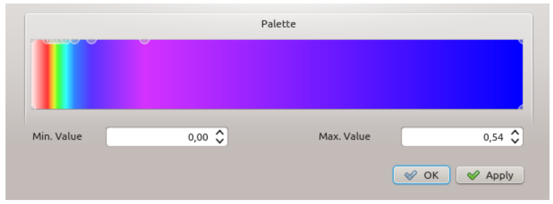


Figure 6.3: Default curvature palette

6.3 Normal

This operation builds layers containing the surface normal vector at every cell. The output layers can be either a register layer containing three fields for the three vector components or three numeric layers. In any case, vector components are real numbers between zero and one. These values can be used to perform operations with other layers. This operation is located at

Operations > Geometry > Normal Map

That will open the operation dialog on the Terminal tab, see Figure 6.5. This tab includes three inputs:

- Output Layer Type. There are two options: Numeric and Register. In the first case three numeric layers will be created; in the second one only one qualitative layer will be created.
- Result layer Name: This is the base name of the new layer that will be

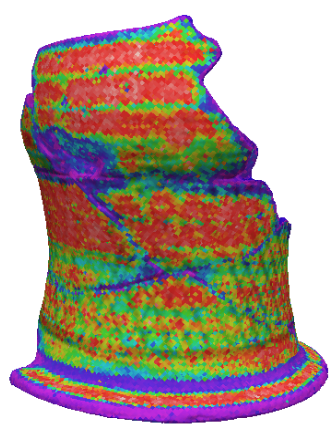


Figure 6.4: Curvature layer

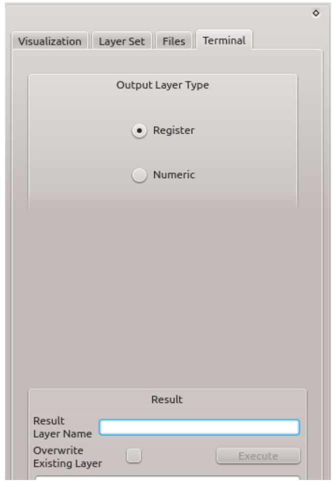


Figure 6.5: Normal map Tab

created. It can be any string. When numeric layers are created the name of the three output layers have suffixed X,Y and Z.

- Overwrite Existing Layer: This box must be checked in order to activate the Execute button when the output layer exits. In this case it will be rewritten.

Figure 6.2 shows the three numeric layers containing the three components of normal vector.

Remark 11. Creating a quantitative normal layer generate a large data base table.

6.4 Orientation

The orientation layer contains, for every cell, the angle between the normal vector at the cell and an input vector given by the user. This function can be used to segment walls and floors on buildings or to estimate the susceptibility to erosion. To run this operation enter

Operations > Geometry > Orientation

This will open the orientation dialog on the Terminal tab, see Figure 6.7. This tab includes, besides the usual *Overwrite Existing Layer* checkbox and the *Result layer Name* input box, three input boxes to enter the component of the reference vector. To get an idea of the reference system you can display the coordinate axis (see chapter 5). The axis colors correspondence is red-X, green-Y and blue-Z (see figure 6.8).

Figure 6.9 shows the orientation layer in relation to the vector (1,1,1) using a modified color palette. Orientation angles are computed in the range (-180,0).

6.5 Roughness

Surface roughness is an important feature of cultural heritage models. It is a measure of the surface texture and is related to the erosion and restoration processes. We define the roughness as the average distance between the tangent sphere and the surface surrounding the cell.

As for other properties, CHISel generates a new layer containing the roughness information. In this case the operation is at

Operations > Geometry > Roughness

This will open the roughness computation dialog on the Terminal tab. This tab contains only the usual *Overwrite Existing Layer* check-box and the *Result layer Name* input box. Roughness is currently computed on the immediate neighborhood of the cell. A more detailed technical description of the computation of the curvature can be found in [Lope13] (in Spanish) .

Roughness values are the mean distances measured using the metric unit entered when creating the model. The roughness of a flat surface or a sphere is zero.

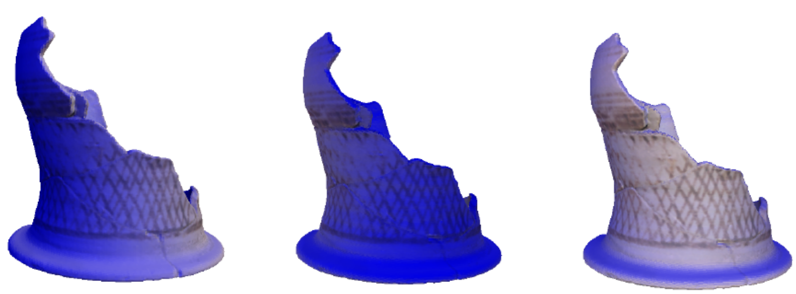


Figure 6.6: Example of normal layers. Left X, middle Y, and right Z.

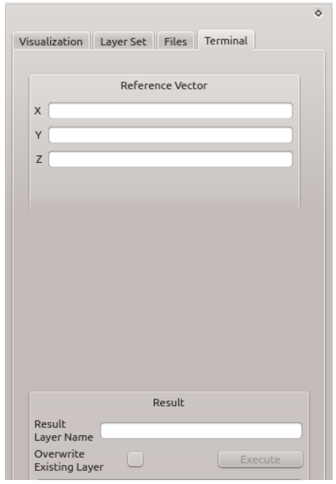


Figure 6.7: Normal map Tab.

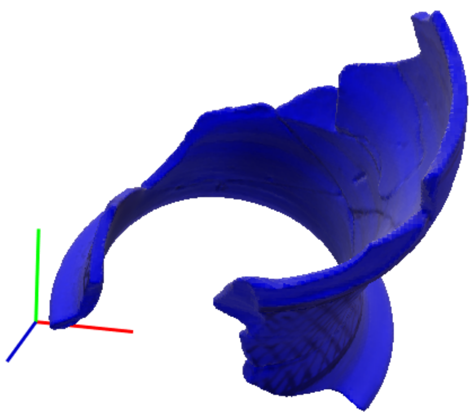


Figure 6.8: Orientation of a model displaying the coordinate axis.

Figure 6.10 shows the roughness layer computed for the vessel model.

6.6 Color

Generated color layers can be stored either as register layers or by using three numeric layers, following a scheme similar to that of the normal vector. In any case, each cell will contain the average RGB cell color. Color components are given as integer numbers in the range of 0 to 255. This operation is located at

Operations > Texture > Texture to Layer

This will open the operation dialog on the Terminal tab, see Figure 6.5. This allows us to chose the output Layer Type, between two options: Numeric and Register. In the first case four numeric layers will be created, while in the second one only one qualitative layer will be created. Numeric layers use the same base name and suffixes A (for alpha), R,G and B for the color components.

Figure 6.11 shows the three numeric layers containing the three color components.

Remark 12. Small artifacts can be seen in figure 6.11 due to the use of Nexus to render the model. Information that is stored in these small cells are not drawn. This problem will be solved in the next release.

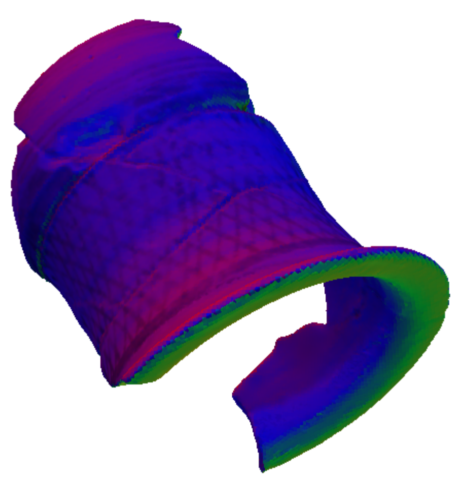


Figure 6.9: Example of orientation computation.

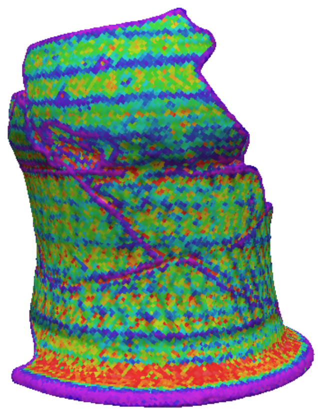


Figure 6.10: Example of roughness layer.

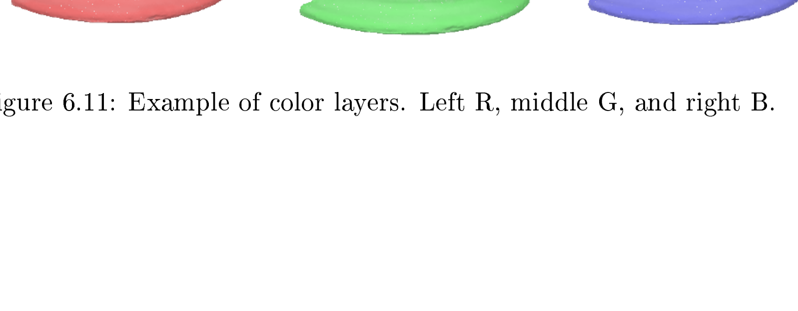


Figure 6.11: Example of color layers. Left R, middle G, and right B.

Chapter 7

Operations

Attribute layers can be combined and transformed, generating new layers. Operations are used to analyze the information concerning the artifact.

CHISel can apply multiple types of operations on its layers. Usually these operations take one or more input layers and parameters (numeric values or expressions) and produce new layers as a result. Although these operations only work on numeric layers, the system has functions for transforming register layers into numerical ones. See section 3.5.

Remark 13. Most operations have an execute button and an overwrite check-box. The execute button is activated when the output layer name is entered (and the return key is pressed) and when the overwrite check-box is selected (if the output layer exists).

Our system prototype uses the following operations to analyze the information:

- **Distance field.** This operation generates a layer containing the distance to the non-null cells on the input layer through the object surface.
- **Null value manipulation.** This operation allows CHISel to manage cells with no value (null cells) in two different ways:
  - Convert null to value: with this mode, the operation assigns a value given by the user to the null cells of the input layer.
  - Convert value to null: with this mode, the operator transforms every cell of the input layer whose value matches the value specified by the user into a null cell.
- **Map algebra.** This operation compute a new layer by performing arithmetic and logic operations on numeric layers. The user introduces an algebraic expression containing arithmetic operators, mathematical functions and logic operators on existing layers and CHISel generates a new layer. The syntax of the expression is that of GRASS module r.mapcalc,

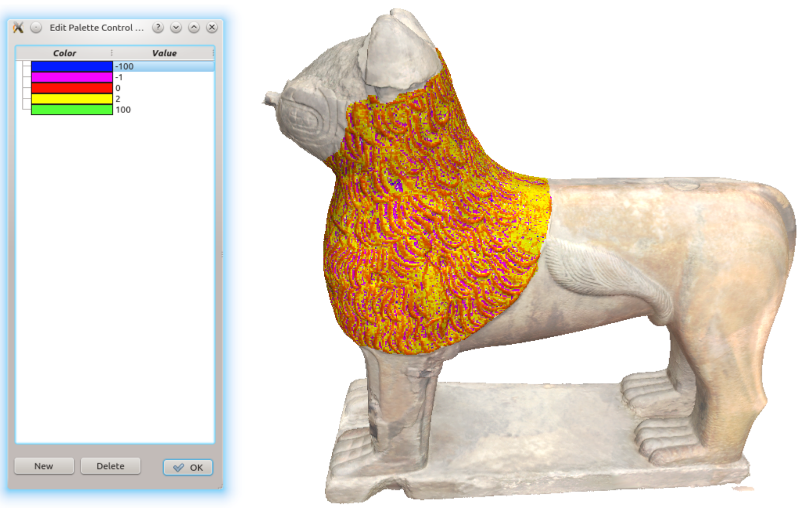


Figure 7.1: Example of algebraic operation

which is very powerful[?]. The operation is computed cell by cell. As an example, the expression:

$$index = if(segment == 3, sqrt(curvature)/roughness, null())$$

computes a new layer whose cells contains the square root of the curvature layer value divided by the roughness layer value on the area with value 3 on segment layer, and null outside this area. The result is shown in Figure 7.1.

- **Neighborhood analysis**. The Neighborhood operator computes a value for each cell as a function of the values on its neighboring cells. CHISel can compute four different functions: Maximum, Minimum, Average and Standard Deviation. The computation is performed on the immediate neighborhood of the cell (at a distance of one cell).
- **Reclassification.** This operation applies a function to the attribute values (in other words, it performs a recodification of the attributes).
- **Resolution adjustment**. This operation can modify the layer resolution (resolution can be increased by interpolating the information, or reduced by resampling).

The following subsection explains these operations.

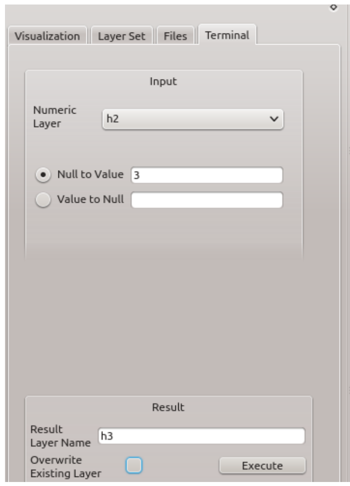


Figure 7.2: Null operation widget

7.1 Null values manipulation

This operation allows us either to assign a value to null cells or assign a null value to non null cells. It only woks with quantitative layers. It is located at

Operations > Null Values

When the operation is performed the widget shown in Figure 7.2 is displayed on the terminal tab. You must select the input layer, the operation to be performed and the output layer name.

Remark 14. It is not possible to perform the two modifications at the same time (null to value and value to null). If you need to do so, you must execute the operation twice. In the worst case you will need three operations if you want to interchange the values: change null value to a non used category, then change the desired value to null, and finally the non-used category to the desired category.

Figure 7.3 shows the result of the replacement of the null values on the left layer by value 3 (The palette of the result layer has been adjusted to allow comparison).

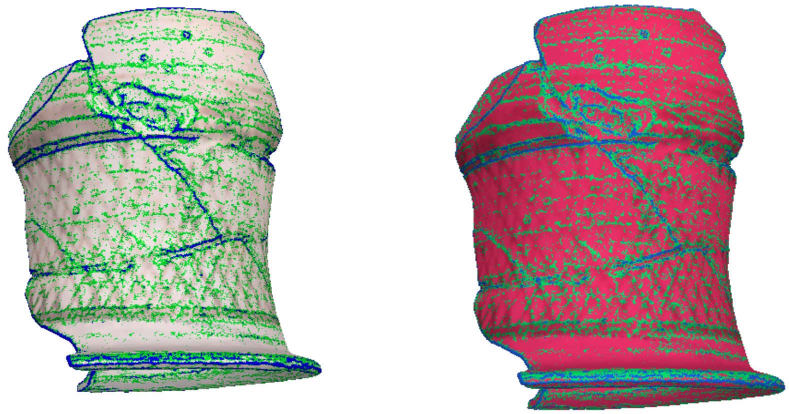


Figure 7.3: Example of null operation. Original left, result right

7.2 Map algebra

Map algebra is a powerful quantitative layer manipulation mechanism. It uses the syntax of r.mapcalc operation from GRASS GIS [Nete08]. This operation is located at

Operations > Arithmetic - Logic

Invoking it changes the terminal tab display to the mapcalc widget shown in figure 7.4.

The input expression is used to introduce the formula to apply (see section 7.2 for a description of the syntax). Note that the output layer name is part of the expression, so the usual output name input that is shown at the bottom is not used. Figure 7.5 shows the result of computing a new layer using the expression:

$$h2 = if(roughness > 0.015, 2, if(roughness > 0.005, 1, null())) \tag{7.2.1}$$

This expression assigns value 2 to cells with values on layer roughness larger than 0.015, 1 to cells with values between 0.005 and 0.015, and null to all other cells.

Figure 7.4: Null operation widget

Expressions are computed cell by cell. That is, the same formula is applied to all the cells. When the formula can not be computed for a cell, the cell is assigned a null value. This may happen for arithmetic errors or when some of the operand layers have a null value at a cell.

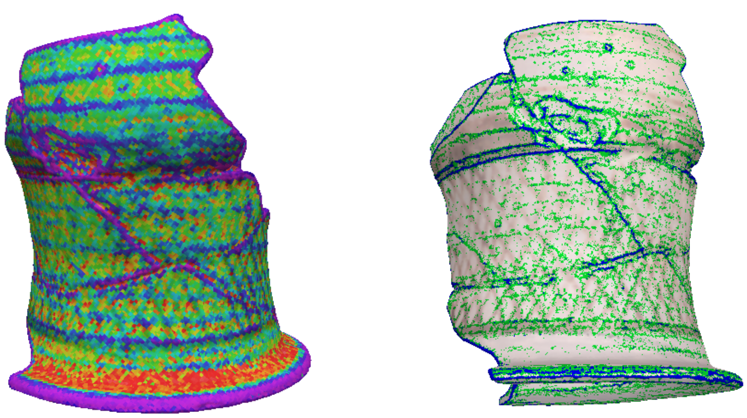


Figure 7.5: Mapcalc example. Roughness layer at the left; h2 layer at the right, using expression 7.2.1

Mapcalc syntax

This section contains a summary of the syntax. For a detailed description of the syntax please read the r.mapcalc documentation at GRASS GIS site (http://grass.osgeo.org/grass64/manuals/r.mapcalc.html).

The expression is an assignment operation with the new layer name on the left and equal sign and the formula to the right. The expression is a formula that uses layer names as variables and may contain arithmetic operators:

- + Addition
- **-** Subtraction
- / Division
- ***** Multiplication
- **%** modulus
- ^ exponentiation

Logical operators

- **&&** And
- || Or
- **!** Not

Comparison

- > Greater
- >= Greater or equal
- < Smaller
- <= Smaller or equal
- == Equal
- **!**= Not equal

Expressions may contain also parenthesis and decisions

- **if**(exp, A, B) if exp evaluates to true then A is computed else B is computed

And they may also contain functions. Some of the most useful are[1]

- **abs**(x) return absolute value of x

[1] Source: http://grass.osgeo.org/grass64/manuals/r.mapcalc.html

- **acos**(x) inverse cosine of x (result is in degrees)
- **asin**(x) inverse sine of x (result is in degrees)
- **atan**(x) inverse tangent of x (result is in degrees)
- **atan**(x,y) inverse tangent of y/x (result is in degrees)
- **cos**(x) cosine of x (x is in degrees)
- **double**(x) convert x to double-precision floating point
- **exp**(x,y) x to the power y
- **float**(x) convert x to single-precision floating point
- **isnull**(x) check if x = NULL
- **log**(x) natural log of x
- **log**(x,b) log of x base b
- **max**(x,y[,z...]) largest value of those listed
- **median**(x,y[,z...]) median value of those listed
- **min**(x,y[,z...]) smallest value of those listed
- **not**(x) 1 if x is zero, 0 otherwise
- **null**() Generates a null value
- **pow**(x,y) x to the power y
- **round**(x) round x to nearest integer
- **sin**(x) sine of x (x is in degrees)
- **sqrt**(x) square root of x
- **tan**(x) tangent of x (x is in degrees)
- **xor**(x,y) exclusive-or (XOR) of x and y

7.3 Distance fields

A distance field is a layer containing the distance to the nearest non-null cell of a reference layer at every cell. The distance is measured through the surface. This operation is located at

Operations > Surface Distance

Invoking it changes the terminal tab to the widget shown in figure 7.6.

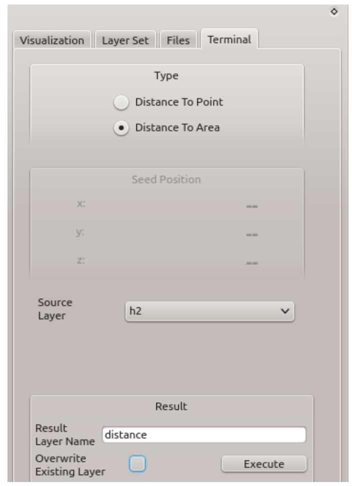


Figure 7.6: Null operation widget

Click on Distance to Area radio button (otherwise the system will compute the distance to a single point).

Select the source layer, which must be quantitative layer, gives the output layer name and click the execute button.

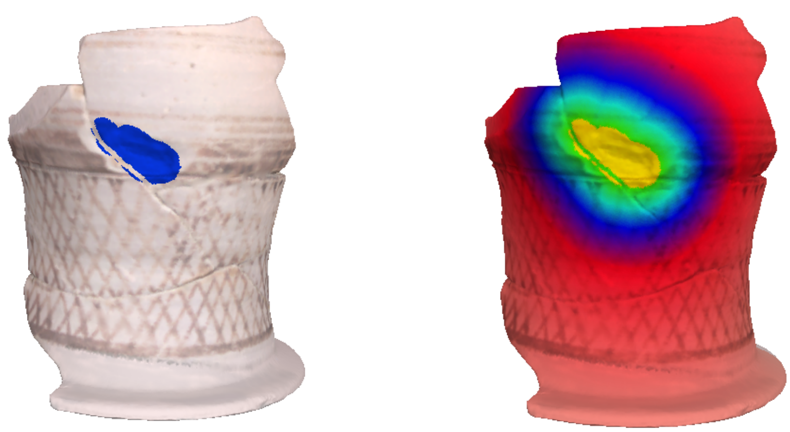


Figure 7.7: The image of the layer on the right shows the distance map to the layer on the left.

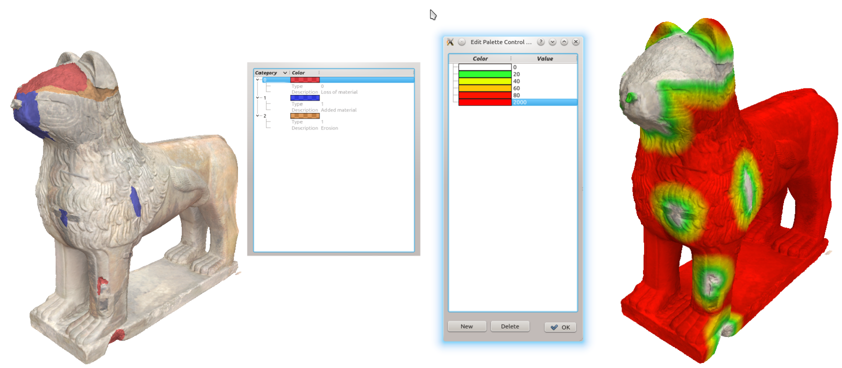


Figure 7.8: Distance field on the sculpture of a Lion from the Alhambra.

Figure 7.7 right shows the distance field computed from the layer shown on the left (the color palette has been adjusted). As another example, figure 7.8 shows the distance field with the palette used as legend.

7.4 Neighborhood analysis

Neighborhood analysis is a powerful operation that can be used to perform computations based on the topological structure of the artifact. It computes a new layer whose values are computed as a function of the values on the surrounding area of every cell. It is located at

Operations > Statistic > Neighborhood Estimator

When the operation is performed, the widget shown in Figure 7.9 is displayed on the terminal tab. You must select the input layer, which must be a quantitative layer, the analysis operation to be performed (called Estimator), which can be

- Maximum
- Minimum
- Average
- Variance
- Standard deviation

and the name of the output layer.

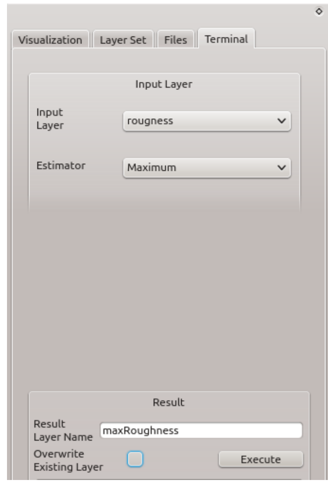


Figure 7.9: Neighborhood Analysis widget

Figure 7.10 shows the result of the computation of the maximum of the roughness layer (middle image), the palette has been adjusted to allow comparisons (saving the palette of the original layer and then loading it to the new one).

Computation is performed considering only direct neighbor cells. You can generate a computation taking into account a larger neighborhood using this

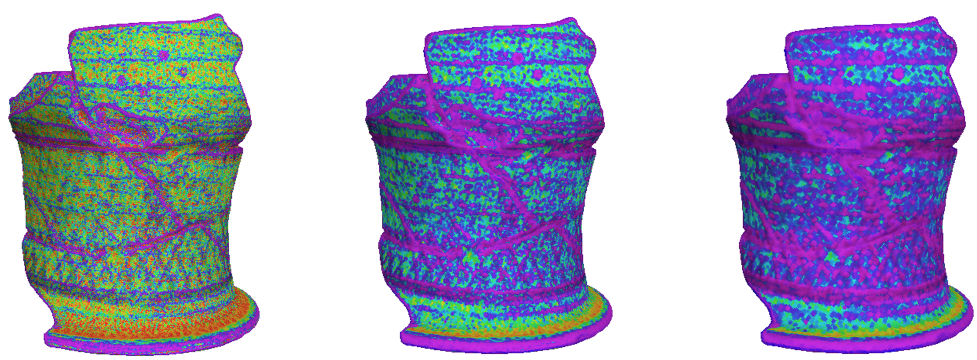


Figure 7.10: Example of neighborhood analysis. From left to right: original layer and two consecutive execution of maximum operation

tool several times. Figure 7.10 shows, on the right image, the computation of the maximum of the previously computed layer.

Chapter 8

Developing a project

The main task when designing a model for a specific application is to design the initial set of layers to be used. This must be done taking into account three restrictions:

- A layer stores the data related to just one attribute. That is, layers must not contain the Cartesian product of two different properties. The data can be quite complex, but there must be only one property. The property can be a chemical analysis, whose data contains several numbers and a spreadsheets.

- A layer assigns just one attribute value to each cell. If we have more than one attribute value obtained at different times, each one will be stored as a different layer.

- Layer data and layer definition must be consistent. Layer data must be directly related to layer definition.

Do not panic. You can add a new layer at any moment.

Sometimes it may not be obvious what must be considered as an attribute. The second condition forces us to redesign a layer if it could have more than one value at the same place. As an example, suppose it is necessary to store information about erosion and composition. We can assign a data base record to our cells containing its erosion index and composition code, but in this way the layer will contain two properties. These complex layers are more difficult to create, edit and manipulate. These two fields must be stored using two different layers, which could be rendered together if needed. Moreover it is possible to generate the layer containing the Cartesian product using layer composition operations, but this layer must be used as a derived information that will be generated from the master normalized layer. One approach to designing the layers set is to factorize the data following a decomposition process so that the resulting attributes cannot be split into two simpler ones that have different mapping on the model surface.

Finally, do not forget that this is a research prototype, it can fail. Save your data frequently and generate backup copies of the model directory.

8.1 Needing help?

If you decide to go on using the system, we will be right there to help you. Please send any doubts comments or criticism by email to lrv@ugr.es, or by phone to (+34) 645 885 167.

And, of course, we will appreciate any feedback you can provide us to improve the next version of the system.

Chapter 9

Menu structure

This appendix contains the whole structure of the menus.

- File
  - Load Model: Read a CHISel model (Chapter 2).
  - Save Model: Save current model (Chapter 2).
  - Save Model as: Save current model in a new directory (Chapter 2).
  - Import Model: Read a PLY file and create a CHISel model (Section 2.1).
  - Export Model: Create a PLY containing the mesh and the current displayed color.
  - Exit: Exit the application.
- Query
  - View Model info: Display global information of the CHISel model.
  - Rulers: Performs distance measurement (to select the points use control and left mouse button).
  - Get Value: Retrieve the information associated in the selected layer at a point (Section 4.1), press escape to exit.
  - Layer Area Stats: Report area covered by every category of the selected layer (Section 4.3).
  - Database Query: Performs a query using a SQL command (Section 4.2).
- Visualization
  - Capture Image: Generate a scree-shot of the rendering area (Section 5.2).

  - Use Model Color: Draw the model using the original color contained in the PLY file. Layer color will be draped over it.
  - Show Model BB: Display the bounding box of the model.
  - Show Grid: Toggle the rendering of the ground grid.
  - Show Axis: Toggle the visualization of the axis.
  - Show Spatial Index Cells: Toggle the visualization of the cells used to index the model . Activating it make rendering slower.
  - Voxelized Mesh: Toggle the visualization of the portion of the surface contained in every cell using a chessboard layout. Activating it make rendering slower.
  - Fragment Classify: Used for debugging purposes.
  - Render
    - Radiance Scaling: Toggle the rendering using radiance scaling, that can be useful to detect discontinuities on the surface.
  - Scene Light: Modify light parameters (ambient, diffuse, and specular intensity).

- Tools
  - Selection Box: Used for debugging purposes.

- Option
  - Set Metric Units: Modify the distance unit used. Modifying it does not scale the model, use this option only when the unit specified when imported the model was not correct (Section 2.2).
  - Resolution: Modify working resolution. All new layers will be created using the specified resolution.
  - Model LOD Config: Configure multiresolution representation using NEXUS.
  - View Console: Show the terminal tab.
  - Binary Layers: Toggle the representation of layer as binary files. When it is selected all layers will be saved using binary representation that can be more compact.

- Layers
  - New Layer: Create a new layer (Chapter 3).
  - Edit Layer: Allow to modify attributes associated to the layer (Section 3.2).
  - Delete Layer: Delete a layer (Section 3.2).
  - Copy Layer: Copy a layer (Section 3.2).

- Modify Table Layer: Modify the structure of the database table associated with a layer (Section 3.4).

- Categories
  - Create Category: Create a new category value for the selected layer (3.2).
  - Edit Category: Edit category values for the selected layer (3.2).
  - Delete Category: Delete a category value for the selected layer (3.2).

- Palette
  - Edit Ctrl Points: Modify the palette for the selected layer using control points (Section 3.3).
  - Edit Intervals: Modify the palette for the selected layer using intervals definition (Section 3.3).
  - Load: Load a palette from disk and associate it to the selected layer (Section 3.3).
  - Save: Save the palette associated to the selected layer to a disk file (Section 3.3).

- Operations
  - Arithmetic-Logic: Computes new layers defined by a mathematical formulae (Section 7.2).
  - Geometry
    - Normal Map: Computes a layer that contains the normal to the surface (Section 6.3).
    - Curvature: Computes a layer that contains the curvature of the surface (Section 6.2).
    - Roughness: Computes a layer that contains the roughness of the surface (Section 6.5).
    - Orientation: Computes a layer that contains the orientation of the surface with respect to a given vector (Section 6.4).
  - Texture
    - Texture to Layer: Generate a layer that contains the surface color (Section 6.6).
    - Layer to Texture: Compute a new texture for the object from the specied layers.
  - Conversion
    - Register to Scalar: Create a new scalar layer that contains information obtained from a register layer (Section 3.5).

        - ◇ Scalar to Register: Create a register layer that contains the information of an scalar layer (Section 3.5).
    - ○ Surface Distance: Compute a layer containing the distance to the non null cells of a given layer (Section 7.3).
    - ○ Null Values: Manage null values in an scalar layer (Section 7.1).
    - ○ Clipping: Computes a new layer clipping information from a specified layer.
    - ○ Resampling: Create a new scalar layer re-sampling a layer.
    - ○ Reclassification: Create a new scalar layer applying reclassification rules to a layer.
    - ○ Statistic
        - ◇ Neighborhood Estimator: Computes a layer that contains at each cell the result of an statistical function applied to the cells at its neighborhood (Section 7.4).
        - ◇ Multivariate Statistics: (Section 4.4).

- Help
    - ○ Manual: Link to this manual.
    - ○ About: copyright notice.
    - ○ Report: Information on how to report an error.

Bibliography

[Nete08] M. Neteler, H. Mitasova, Open Source GIS: A GRASS GIS Approach Springer, New York (2008) 48

[Lope13] L. López, J.C. Torres, G. Arroyo: Measuring Surface Roughness on Cultural Heritage 3D models. CEIG - Spanish Computer Graphics Conference. Published at EUROGRAPHICS DIGITAL LIBRARY. Madrid, September 2013 2, 36, 40

[Sole12] F. Soler, J.C. Torres, A.J. León, M.V. Luzon: Design of an Information System for Cultural Heritage. Spanish Computer Graphics Conference, 113-122 Published at EUROGRAPHICS DIGITAL LIBRARY (2012) 1

[Sole13] F. Soler, J.C. Torres, A.J. León, M.V. Luzon: Design of Cultural Heritage Information Systems based on Information Layers. Journal on Computing and Cultural Heritage, 6(4): 15:1-15:17. (2013)

[Torr12] J.C. Torres, L. López, C. Romo, F. Soler: An information system to analize cultural heritage information. Progress in Cultural Heritage Preservation, 809-816 (2012) 1, 2, 11

[Torr12a] J.C. Torres, F. Soler, P. Cano, A. León, V. Luzón, D. Martín: Sistema de Información para Patrimonio Histórico. Virtual Archaeology Review, 2012, 3 (5), 53-57 (2012) 1

[Torr13] J.C. Torres, L. López, C. Romo, G. Arroyo, P. Cano, F. Lamolda and M. M. Villafranca: Using a Cultural Heritage Information System for the documentation of the restoration process. Digital Heritage 2013. Published at IEEE in Xplore digital library. Marsiella, 28/10-1-11 (2013) 2, 3

[Torr13a] J.C. Torres, C. Romo Título: Análisis espacial de modelos 3D: Simposio Iberoamericano de Análisis Espacial en Arqueología. Santander. 21-23, Octubre 2013 3, 4

[1]

Index