

Determining geometric primitives for a 3D GIS: easy as 1D, 2D, 3D?

Britt Lonneville¹, Cornelis Stal¹, Berdien De Roo¹, Alain De Wulf¹ and Philippe De Maeyer¹

¹*Department of Geography, Ghent University, Krijgslaan 281 (building S8), Ghent, Belgium
{Britt.Lonneville, Cornelis.Stal, Berdien.DeRoo, Alain.DeWulf, Philippe.DeMaeyer}@Ugent.be*

Keywords: 3D GIS, Geometric Primitive, Computational Geometry, Point, Surface, Solid.

Abstract: Acquisition techniques such as photo modelling, using SfM-MVS algorithms, are being applied increasingly in several fields of research and render highly realistic and accurate 3D models. Nowadays, these 3D models are mainly deployed for documentation purposes. As these data generally encompass spatial data, the development of a 3D GIS would allow researchers to use these 3D models to their full extent. Such a GIS would allow a more elaborate analysis of these 3D models and thus support the comprehension of the objects that the features in the model represent. One of the first issues that has to be tackled in order to make the resulting 3D models compatible for implementation in a 3D GIS is the choice of a certain geometric primitive to spatially represent the input data. The chosen geometric primitive will not only influence the visualisation of the data, but also the way in which the data can be stored, exchanged, manipulated, queried and understood. Geometric primitives can be one-, two- and three-dimensional. By adding an extra dimension, the complexity of the data increases, but the user is allowed to understand the original situation more intuitively. This research paper tries to give an initial analysis of 1D, 2D and 3D primitives in the framework of the integration of SfM-MVS based 3D models in a 3D GIS.

1 INTRODUCTION

As a result of increasing computer speed and capabilities and improving acquisition techniques such as laser scanning and photo modelling, 3D models are becoming more and more common in several fields of research (Arav et al., 2014; Siebert & Teizer, 2014; Vanneschi et al., 2014). Consequently, the need arises to use these models for more than just documentation purposes. The idea of integrating these models in a 3D GIS (geographic information system) has already been the subject of an active debate (De Roo et al., 2014; Heras Barros, 2014; Frank, 2008; Wu et al., 2008; Zlatanova et al., 2002). Such a 3D GIS would greatly contribute to both micro- and macroscale research and would allow researchers to perform 3D queries and study a site even after its destruction (e.g. following an archaeological excavation). However, GIS vendors do not seem too eager to implement 3D functionality in their software as this would require a considerable investment in the development of such 3D functions, whereas the economic benefits of this effort have not yet been shown. Most plug-ins for existing GIS software are limited to 2.5D representations (i.e. using 2D primitives in 3D space), while other attempts at the integration of 3D geometry and

semantics, such as the development of the CityGML standard, are focused on specific use cases (cities, ...) and are hard to tailor for every type of acquired data (e.g. point clouds). Moreover, the integration of GIS and CAD/BIM, which have been supporting 3D data for a long time, proves to be difficult (Hijazi et al., 2010). This has led researchers to alternative solutions such as the use of game engines and web GIS (von Schwerin et al., 2013; Rua & Alvito, 2011).

This paper aims to contribute to the development of a 3D GIS by examining different ways of representing 3D data from a geometrical point of view. More specifically, it attempts to give an initial insight into different possible geometric primitives for such a 3D GIS and their specific advantages and drawbacks.

2 GEOMETRIC PRIMITIVES

In order to create or import data in a GIS, a certain geometric primitive has to be selected. Such a primitive is defined by rules on the conceptual representation of a feature, attributes and relation to other features in the data set. Hence, the chosen primitive strongly influences the visualisation of the

data, the topology and the possible data manipulation and querying options. Current 2D vector GIS use both 1D (point) and 2D (line and polygon) primitives. The 3D models constructed out of these primitives are respectively point clouds, wireframe models and meshes, such as TINs (triangular irregular networks). 2D raster GIS allow the user to import raster images into the system, giving every pixel a specific value (RGB, elevation,...). Their 3D counterparts are voxels (volumetric pixels).

When dealing with a 3D GIS, 3D primitives – which function as building blocks for volumetric 3D models – should also be considered. In this respect, Arens et al. (2005) discuss various possible 3D primitives for a geo-DBMS: tetrahedra, polyhedra, polyhedra combined with spherical and cylindrical patches and CAD objects. These last two primitives render CSG (constructive solid geometry) models, as described by Ghali (2008). Another 3D primitive which might be considered for implementation in a 3D GIS is the cube, rendering voxel models. This primitive, which is a specific type of polyhedron, is currently being used mainly for modelling in medical applications and game environments such as Minecraft (Radua et al., 2014; Pasciak & Erwin, 2009). In order to reduce storage capacity and computation time, these voxels can be joined in an octree model (Abdul-Rahman & Pilouk, 2008).

Consequently, the potential geometric primitives can be classified based on their number of dimensions: 1D, 2D or 3D. These nD features lead to respectively point-based, surface-based or solid-based models in a 3D space, a classification that is also introduced by Pouliot et al. (2006) and is shown in Figure 1.

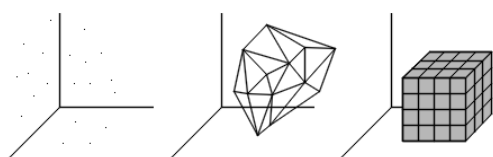


Figure 1: Point-based, surface-based and solid-based models (point cloud, wireframe and voxels).

2.1 1D primitives

Regarding 1D primitives, only one building block is possible, namely points. The resulting point clouds can have various sources. They can be the result of discrete total station or GNSS measurements, a laser scanning operation or the Structure from Motion and Multi-View Stereo (SfM-MVS) process that is often

adopted in photo modelling. Moreover, the points can depict a variety of objects, ranging from characteristic marks on buildings to a complete surface.

By importing point clouds in a 3D GIS, the user would be able to link database data to the distinct points and perform extensive semantic and geometric analyses on these points. This can be exemplified by an excavation where the archaeologist examines the proximity of certain objects (e.g. shards) to investigate their coherence and origin. The main advantage of using point clouds is that the discrete points represent measured values and the data has normally not yet been generalised, giving researchers the chance to investigate the data in their ‘purest’ form.

Moreover, there are already several (free/open source) software that allow point cloud analysis to a certain extent, such as CloudCompare (Figure 2), AutoCAD Civil 3D and Point Cloud Library. However, they occasionally consider the point cloud as a whole and do not allow the user to manipulate or query one single point as is the case in a regular GIS. Another alternative in this respect are some databases that support 3D point coordinates, such as PostgreSQL (with spatial extension PostGIS), Oracle Spatial and MySQL. They might provide researchers with basic tools for analysis but lack the visualisation possibilities of a full-fledged GIS.

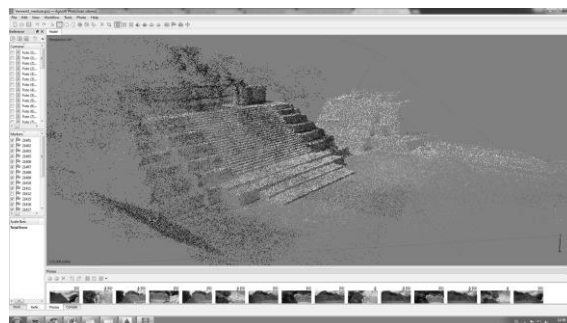


Figure 2: Visualisation of point cloud model of Mayan temple in Agisoft PhotoScan.

2.2 2D primitives

Similar to 2D GIS, 3D GIS adopt two types of 2D primitives, namely lines and polygons, resulting in wireframe models and meshes, such as TINs. These models often find their origin in the manipulation of point clouds. The Delaunay triangulation, for example, is a well-known algorithm for the creation of a triangular mesh out of a point cloud (Cheng et al., 2013). The algorithm tends to maximize the

angles of the resulting triangles and is often used in computational geometry. In comparison to other types of polygons, using triangles as a primitive increases the realism of the final model and allows the reconstruction of complex structures. TINs (2.5D) and meshes (3D) thus succeed in approaching the original shape and look of the modelled object better than point clouds, as the user can intuitively understand the context that the model was created in.

When importing meshes or wireframes into a 3D GIS, several issues arise, the most important one being the connection to attribute data. It should be possible to link data to the object as a whole, but also to a subset of triangles or even one single triangle. This depends among others on the size of the object and the goal of the research. When examining large structures, such as buildings, there should be a way to distinguish several discrete parts of the building, such as windows and floors, and link different attribute data to each part. In other cases, it can be necessary to treat the object and its constituting primitives as a whole, with separate objects having their own separate attribute data. When every triangle is seen as a distinct feature, the software should be aware of the coherence of different triangles and the structures they make up. In 2D GIS, these relationships are defined through the use of topology. A similar system should thus be applied when developing a 3D GIS.

One of the main advantages of using 2D primitives is that this kind of representation is already being used very often in non GIS-related applications and software (e.g. Meshlab, Figure 3). Not only are there several file formats that the representation of meshes (and when necessary store texture), various systems also support these formats and allow users to quickly import them and visualise or edit the meshes that they contain.

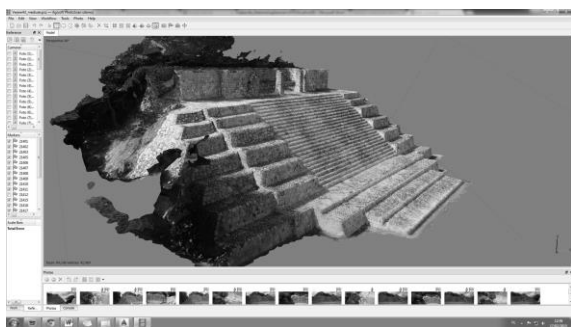


Figure 3: Visualisation of triangular mesh model of Mayan temple in Agisoft PhotoScan.

2.3 3D primitives

When deciding on a suitable 3D primitive, several choices can be made depending on the source data and the research goal. For example, when only data about specific characteristic points of a given object are provided, the researcher can reconstruct the object using CAD objects as is the case with CSG models (Figure 4). When a laser scanning point cloud is provided, the researcher might opt to transform the input data into a polyhedron or tetrahedron model. However, the algorithms that transform the point cloud into these models are more complex and less widespread when compared to those that transform point clouds into a set of 2D primitives.

Whether or not the resulting model depicts the object realistically depends on two factors: the input data and the chosen primitive. When only a few distinct points are available, CSG models suffice as they are able to incorporate all points and reduce the resulting complexity. However, when an entire point cloud is provided through laser scanning or photo modelling, basing the reconstruction on CAD objects would either be inadequate, as this would reduce the accuracy and realism of the model, or result in unnecessarily complex models.

The querying possibilities of these 3D models depend on the chosen primitives as well. The same issues arise when trying to link attributes to a tetrahedron model as when trying to link attributes to a triangular irregular network. As the tetrahedra are calculated primarily with the goal to match the input data, it might be hard to distinguish distinct features in the final model (such as a building's floors, windows, etc.). A segmentation or feature detection algorithm might be required to meet this goal. A CSG model, on the other hand, is built up of well-defined features which were decided on by the user on beforehand. Using these models might simplify the connection to attributes and thus the integration into a 3D GIS. Nevertheless, it should be taken into account that the model might not represent the input data completely truthfully. Ideally, both a realistic representation of the object and a straightforward connection to its defining attributes should be acquired when using a certain primitive. Considering validation, realism, modelling and algorithms, Arens et al. (2005) prefer polyhedra as 3D primitives when developing a 3D geo-DBMS.

However, the main drawback of currently using 3D primitives is that there are far less possibilities compared to 1D and 2D primitives. While laser

scanning and photo modelling software often offer users the possibility to transform point clouds into meshes or wireframes, this is not the case for any of the above 3D primitives. Moreover, there are less visualisation and editing software available that support file formats containing these kinds of primitives. Most applications handling volumetric or solid models are used in medical imagery analysis (e.g. MeVisLab and VoluMedic), the gaming industry (e.g. Blender, Unity) or CAD software (e.g. AutoCAD). Furthermore, there are some open source toolkits and derived programs (e.g. VTK and ParaView) which enable volumetric methods.

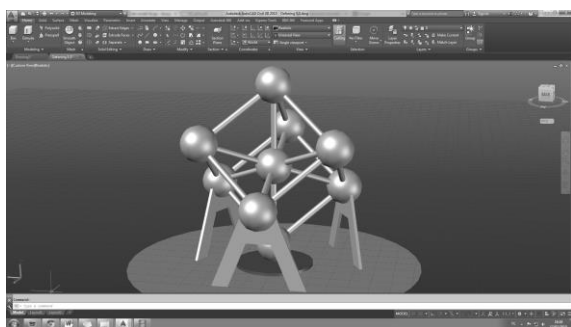


Figure 4: Visualisation of volumetric CSG model of Atomium in AutoCAD.

3 DISCUSSION

Both one-, two- and three-dimensional primitives in a 3D space have been elaborated in the previous section. Based on the discussed characteristics, Table 1 is composed. It evaluates three main aspects of every primitive: how well/truthfully it represents or visualises the real-world object, what the possibilities are when implemented in a 3D GIS (e.g.

1:1 relation object-attributes?) and how well the primitive is embedded in current practice or 3D applications. On the one hand, 2D primitives seem the best fit in many common analyses, due to their representation possibilities and acceptance in a wide variety of applications. However, they seem to lack particular qualities that are necessary when performing GIS analyses. Certain 3D primitives, on the other hand, are promising when it comes to their implementation in a 3D GIS, but are hardly supported and only used in very specific cases. 1D primitives, which represent the object in its most simplified form, have the benefit of already being available in current geo-DBMS and being supported by several file formats and software, but have very limited non-spatial querying possibilities and might fall short where data visualisation is concerned.

How to decide on what primitive to use mainly depends on the research goal. Research focused on the (truthful) representation of objects will probably benefit most from a mesh model and will thus use a series of 2D primitives in a 3D space (Table 1), whereas research that involves volume calculations and a volumetric representation of the object should consider the use of a fully 3D primitive. Even when a certain dimensional complexity is selected, a choice has to be made regarding the conceptual model of that primitive. The preference of one type of primitive over the other is of vital importance to the further course of the research and will depend not only on the input data, but also on the desired output and available software.

However, it is of significant importance that GIS vendors see the necessity of the implementation of 3D models in their software and consequently provide users with various tools to visualise, manage and analyse these models. Common GIS file formats, such as ESRI's shapefile, and open standard geospatial formats, such as GeoJSON, should

Table 1: Overview of potential geometric primitives – advantages and drawbacks.

Dimensions	Primitive	Representation	3D GIS	Current situation
1D	Point	+/-	+/-	+
2D	Line Polygon	+	+/-	+
3D	Tetrahedron Polyhedron Polyhedron combined with spherical and cylindrical patches CAD objects	+	+/-	+/-

incorporate not only 1D and 2D primitives, but also 3D primitives, the possibility to add 3D coordinates to 1D and 2D primitives and the support for analysing nD primitives in a 3D space. The open source community could speed up this process by either creating their own format, by backing this evolution or by modifying the GIS possibilities of current 3D modelling formats. Some of the most common file formats in this respect are Collada (.dae), Wavefront OBJ (.obj) and Polygon File Format (.ply). Moreover, the Virtual Reality Modeling Language (.vrl) and Extensible 3D (.x3d) format are being recognised as ISO standards. These formats can be imported in mesh editing software such as MeshLab, but as well in game engines and CAD software (e.g. Blender, AutoCAD). Most formats, however, do not support attribute data or even the explicit definition of a coordinate system.

It can also be questioned if a new, 3D GIS-specific file format should support every possible 3D primitive or limit itself to one specific type of 3D primitive. This would of course narrow the possibilities of such a format, but might also have the advantage of being straightforward, whereas the opposite might cause confusion when a specific set of operations is available for one primitive but not for another.

Another concern should be the lifespan of such a format. It is important that this format is accepted by the GIS community in order for it to be successful and used over a large period of time. This is closely connected to the chosen primitive and can thus influence the preference for one primitive over the other.

4 CONCLUSIONS

As a result of the rising interest in 3D models, the need for the development of a 3D GIS increases. Before such a system can be conceived, several initial issues have to be tackled, such as the decision on a geometric primitive through which the models will be imported into the software. 1D (point) and 2D (line and polygon) primitives are well known and their use in both 2D and 3D applications is widespread. However, they seem to lack certain qualities that influence both the visualisation of the objects that they represent and the implementation and analytical possibilities of these objects in a 3D GIS.

Consequently, some thought should go into the implementation of 3D primitives, and their possible integration into standard GIS file formats. These 3D

primitives can form either tetrahedral, polyhedral or CSG models (using CAD objects). All of these primitives have certain advantages and drawbacks and the preference for one primitive over the other is based not only on the desired outcome, but also on the available input data (characteristic points vs extensive point cloud). 3D primitives have the advantage that they allow volumetric representations and operations and thus show the object in the same way as it is also perceived in the real world.

It is thus of great importance that GIS vendors see the necessity of the implementation of 3D models in their software. The open source community can play a vital role in this process, as they are usually at the forefront of technical developments. Implementing 3D primitives into new or existing file formats and software may seem a considerable challenge. However, the possibilities and the progress this encompasses can encourage researchers dedicated to several domains and working on diverse projects to appreciate 3D models in new ways and even rethink the way in which 3D models are conceived.

5 FUTURE RESEARCH

This position paper conveys an initial overview of the possibilities of several geometric primitives in light of the development of a 3D GIS. Future research will focus on the use of 3D geometric primitives (e.g. tetrahedra, CSG objects, voxels,...) in such a GIS. It will thus incorporate two aspects: (1) the conversion of existing 3D point clouds into volumetric 3D models using CAD, BIM and reverse engineering techniques, (2) an extensive overview of the advantages and drawbacks of the applied geometric primitives, focusing on their future use in a 3D GIS and based on both a theoretical and empirical pillar. If possible, the use of open source software will be favoured.

ACKNOWLEDGEMENTS

Financial support from the Special Research Fund from Ghent University and the Research Foundation Flanders (FWO) is gratefully acknowledged.

REFERENCES

Abdul-Rahman, A., Pilouk, M., 2008. *Spatial Data Modelling for 3D GIS*, Springer. Berlin, 1st edition.

- Arav, R., Filin, S., Avner, U., Bar-Oz, G., Nachmias, A., Nadel, D., 2014. Use of terrestrial laser scans for high resolution documentation and 3D modelling of 'desert kites'. In *Near Eastern Archaeology*, vol. 77, no. 3, pp. 219-222.
- Arens, C., Stoter, J., van Oosterom, P., 2005. Modelling 3D spatial objects in a geo-DBMS using a 3D primitive. In *Computers & Geosciences*, vol. 31, no. 2, pp. 165-177.
- Cheng, S., Dey, T., Shewchuk, J., 2013. *Delaunay mesh generation*, CRC Press. Boca Raton, 1st edition.
- De Roo, B., Ooms, K., Bourgois, J., De Maeyer, P., 2014. Bridging archaeology and GIS: influencing factors for a 4D archaeological GIS. Paper presented at 5th International conference on Cultural Heritage (EuroMed 2014), Limassol, 3-8 November.
- Frank, A., 2008. Requirements for 3D in geographic information systems applications. In van Oosterom, P., Zlatanova, S., Penninga, F., Fendel, E. (Eds.) *Advances in 3D Geoinformation Systems*. Berlin Heidelberg: Springer.
- Ghali, S., 2008. *Introduction to Geometric Computing*, Springer eBooks. London, 1st edition.
- Heras Barros, V., 2014. Towards a GIS based monitoring tool for a preventive conservation management of the world heritage city of Cuenca. Unpublished PhD thesis. Katholieke Universiteit Leuven, Leuven.
- Hijazi, I., Ehlers, M., Zlatanova, S., 2010. BIM for geo-analysis (BIM4GEOA): set up of 3D information system with open source software and open specification (OS). In Kolbe, T., König, G., Nagel, C. (Eds.) *ISPRS Archives*, vol. XXXVIII-4/W15, pp. 45-49.
- Pasciak, A., Erwin, W., 2009. Effect of voxel size and computation method on Tc-99m MAA SPECT/CT-based dose estimation for Y-90 microsphere therapy. In *IEEE Transactions on Medical Imaging*, vol. 28, no. 11, pp. 1754-1758.
- Pouliot, J., Lachance, B., Kirkwood, D., 2006. L'importance de la modélisation géométrique 3D lors de l'élaboration d'un SIG 3D: Exemple du développement d'une structure topologique pour une application géologique". In *Revue Internationale de Géomatique*, vol. 16, no. 1, pp. 29-49.
- Radua, J., Canales-Rodriguez, E., Pomarol-Clotet, E., Salvador, R., 2014. Validity of modulation and optimal settings for advanced voxel-based morphometry. In *NeuroImage*, vol. 86, pp. 81-90.
- Rua, H., Alvito, J., 2011. Living the past: 3D models, virtual reality and game engines as tools for supporting archaeology and the reconstruction of cultural heritage – the case-study of the Roman villa of Casal de Feiria. In *Journal of Archaeological Science*, vol. 38, no. 12, pp. 3296-3308.
- Siebert, S., Teizer, J., 2014. Mobile 3D mapping for surveying earthwork projects using an Unmanned Aerial Vehicle (UAV) system. In *Automation in Construction*, vol. 41, pp. 1-14.
- Vanneschi, C., Salvini, R., Massa, G., Riccucci, S., Borsani, A., 2014. Geological 3D modelling for excavation activity in an underground marble quarry in the Apuan Alps (Italy). In *Computers & Geosciences*, vol. 69, pp. 41-54.
- von Schwerin, J., Richards-Rissetto, H., Remondino, F., Agugiaro, G., Girardi, G., 2013. The MayaArch3D project: a 3D webGIS for analysing ancient architecture and landscapes. In *Literary and Linguistic Computing*, vol. 28, no. 4, pp. 736-753.
- Wu, Q., Xu, H., Zhou, W., 2008. Development of a 3D GIS and its application to karst areas. In *Environmental geology*, vol. 54, no. 5, pp. 1037-1045.
- Zlatanova, S., Abdul-Rahman, A., Pilouk, M., 2002. Trends in 3D GIS development. In *Journal of Geospatial Engineering*, vol. 4, no. 2, pp. 71-80.