



Technische Universität München
Institut für Photogrammetrie und Kartographie
Lehrstuhl für Kartographie
Univ.-Prof. Dr.-Ing. Liqiu Meng

Visual reconstruction of archaeological data of the Sanctuary of Diana at Nemi, Italy

Mariana Danielová

Master Thesis

Editing time: 1.04.2014 – 30.09.2014

Course of study: Master of Science in Cartography

Supervisors: Dr.-Ing. Holger Kumke
Dipl.-Ing. Stefan Peters

Cooperation: Archaeologist Dr. Francesca Diosono

2014

DECLARATION OF AUTHORSHIP

I, Mariana Danielová, hereby declare that the thesis submitted is my own unaided work. All direct or indirect sources used are acknowledged as references.

I am aware that the thesis in digital form can be examined for the use of unauthorized aid and in order to determine whether the thesis as a whole or parts incorporated in it may be deemed as plagiarism. For the comparison of my work with existing sources I agree that it shall be entered in a database where it shall also remain after examination, to enable comparison with future theses submitted. Further rights of reproduction and usage, however, are not granted here.

This paper was not previously presented to another examination board and has not been published.

Place and date

Signature

ABSTRACT

The archaeological virtual reconstructions have recently gained large attention from different fields of science. With increasing interests in creating 3D model, deficiencies are appearing and thus the amount of special requirements from the archaeological field is growing. These demands are listed in two major documents: the London Charter and the Seville Charter. Moreover, procedural modeling approach appears as a suitable method for handling the virtual reconstruction together with the archaeological requirements.

The main aim of this thesis is generation of the 3D reconstruction of the Sanctuary of Diana at Nemi, Italy. However with respect to the archaeological requirements, the second aim is expression of the uncertainty of available archaeological knowledge. Firstly, the 3D model is created based on different types of data: the exact geodetic measurements carried out by research group from Technische Universität München; the archaeological interpretation of the excavation of the temple; and literature sources used to get familiar with the Roman architecture. The entire reconstruction is generated using procedural modeling techniques in the CityEngine software. Next, fuzzy logic theory is introduced in order to quantify the uncertainty which is then visualised combining several visualising methods compatible with procedural modeling and CityEngine environment. Finally, the reconstruction results are published and they are accompanied with necessary project documentation.

CONTENTS

1	Introduction	1
1.1	Visualisation of cultural heritage	1
1.1.1	London Charter	1
1.1.2	Seville Charter	2
1.1.3	Tension between veridical and photo-realistic modelling	3
1.2	Possible approaches of modelling	4
1.2.1	Computer Aided-Design (CAD)	4
1.2.2	Building Information Modeling (BIM)	4
1.2.3	Procedural modeling	5
1.3	Fuzzy logic	5
1.3.1	Fuzzy logic approach in archaeological virtual reconstruction	6
1.4	Motivation, Challenges	7
2	Recent studies	8
2.1	Archaeological reconstruction using procedural modeling	8
2.2	Quantifying the uncertainty using fuzzy logic	9
2.3	Visualizing the uncertainty	10
3	Methodology	15
3.1	Description of archaeological site in Nemi	15
3.2	Available archaeological knowledge	16
3.2.1	General description of Roman temple architecture	16
3.2.2	Technical data about excavation in Nemi	18
3.2.3	Information about the Sanctuary of Diana	18
3.3	Procedural modelling in archaeology	21
3.3.1	Description of the CityEngine	21
3.3.2	Workflow for visual reconstruction of cultural heritage	22
3.4	Handling uncertainty and reliability of the 3D model using fuzzy logic	23
3.4.1	Method for calculation of uncertainty	24
3.4.2	Uncertainty visualization techniques	26
4	Visual reconstruction of the Sanctuary of Diana in Nemi	28
4.1	Generation of the 3D model	28
4.1.1	Analysis of design and temple's parameters	28
4.1.2	Modeling of terminate shapes	33
4.1.3	Colors and textures	40
4.1.4	Reconstruction rules	41
4.1.5	Implementation of levels of detail (LoD)	49
4.2	Modeling uncertainty of the 3D reconstruction	50
4.2.1	Calculation of uncertainty	50
4.2.2	Visualization of uncertainty	52
4.3	Results of the reconstruction	56
4.3.1	Publishing the model online	58
5	Discussion	62
6	Conclusion	65
	Acknowledgement	67
	References	68
	Appendix	72

A	Values of the parameters used for the generation of the 3D model	72
B	List of texture used for the reconstruction	74

LIST OF FIGURES

Figure 1	Illustrations of the project Rome Reborn 2.0	8
Figure 2	Previews of the 3D reconstruction of the ancient city Pompeii	9
Figure 3	Detailed reconstruction of one of the highly ornamented buildings in Xkipché	9
Figure 4	Visualization of the uncertainty using pseudocoloring and transparency	11
Figure 5	Visualization of the uncertainty using pseudocoloring, chessboard texture, and intensity of a noise	11
Figure 6	Mapping an attribute value into the 2D circular textures and 3D solid textures	12
Figure 7	Visualizing uncertainty by variation of edge saturation and edge sketchiness	12
Figure 8	Visualizations of different results of the archaeological reconstruction of the chapel in Vojvodina	13
Figure 9	Side by side illustration of the uncertainty modeling generated in CityEngine	13
Figure 10	Visualizing temporal uncertainty using animation	13
Figure 11	Visualization of the uncertainty using extrinsic visual variables	14
Figure 12	Visualizing an attribute value using spherical, vector, and line glyphs	14
Figure 13	Location of the excavations of the Sanctuary of Diana at Nemi, Italy	15
Figure 14	Photograph of the excavations of the Sanctuary of Diana, taken in 2013	16
Figure 15	Vitruvius's illustration of the dimension of the Ionic column	17
Figure 16	Sketch of the excavation area situated in Nemi, Italy	18
Figure 17	Detailed sketch of the excavations of the Sanctuary of Diana at Nemi, Italy	19
Figure 18	Depiction of the different construction phases of the Sanctuary of Diana	19
Figure 19	Five consecutive steps of the temple generation using procedural modeling	22
Figure 20	Workflow for a typical architectural procedural modeling project	23
Figure 21	Example of CGA shape workflow of temple	24
Figure 22	Legend for Green-Yellow-Red scheme and Opaque-Transparent scheme expressing the uncertainty	27
Figure 23	Sketches of the temple's model, and podium dimensions	29
Figure 24	Characteristics of staircase and dimensions of temple's cellas	30
Figure 25	Distribution of walls and explanation of the calculation of the column's height	30
Figure 26	Division of the roof types and demonstration of the pediment structure	32
Figure 27	Structure of the podium and its components	33
Figure 28	Examples of the podium structures that are adjusted to the inner and outer corners	34
Figure 29	Archaeological data for the properties of the column's shaft	34
Figure 30	Properties of the column shaft element created in SketchUp	35
Figure 31	Structures of the column's capital and base components	35
Figure 32	Properties of the cornice, the uppermost part of the entablature	36
Figure 33	Models of entablature including the corner's structures	36
Figure 34	Representation of the pediment elements	37
Figure 35	Adaptation of the slanting sides of geison in pediment	37
Figure 36	Depiction of the bronze roof tiles	38
Figure 37	Terminate shapes of door and window	38
Figure 38	Statues decorating the pediment	39
Figure 39	Terminate shapes of antefix and altar	39
Figure 40	Donated statue placed behind the temple and its elements	40
Figure 41	Examples of the textures used for the temple reconstruction	41
Figure 42	Reconstruction of the statue of goddess Diana	41

Figure 43	Pavement reconstruction with and without rotation of the texture tiles . . .	48
Figure 44	Demonstration of low and medium LoD	49
Figure 45	Presentation of the high LoD	50
Figure 46	Variations of final temple models with their reliability values	51
Figure 47	Uncertainty of parts of the temple expressed by different visualization schemes	53
Figure 48	Swipe tool in CityEngine Web Viewer comparing uncertainty of parts of the temple illustrated by different visualization schemes	53
Figure 49	Visualization of different variations of the Diana temple using color schemes to enhance the value of reliability	54
Figure 50	Using interactivity in order to enhance uncertainty visualization in CityEngine	54
Figure 51	Modifying different parts of the temple using Inspector window in CityEngine	56
Figure 52	Modifying parameters of the roof angle in the Inspector window in CityEngine	57
Figure 53	Changing of the reliability values via the Inspector window in CityEngine . .	57
Figure 54	Changing of the reliability values via the Inspector window in CityEngine . .	58
Figure 55	Sequence of the temple generation, beginning with certain remains on the left and ending with detailed and less certain model of the temple on the right	58
Figure 56	Web page containing all required information about the reconstruction . . .	59
Figure 57	CityEngine Web Viewer, scene containing realistic temple reconstruction . .	59
Figure 58	CityEngine Web Viewer, scene showing the uncertainty of the temple parts expressed by different visualization schemes	60
Figure 59	CityEngine Web Viewer, example of the set of 16 temples and corresponding color legend	60
Figure 60	CityEngine Web Viewer, few examples from the sequence of temple's model generation	61

LIST OF TABLES

Table 1	Reliability values of the objects of the Sanctuary of Diana	25
Table 2	Selection of colors used for the temple reconstruction	40
Table 3	Calculation of uncertainty for the stepwise process of the temple generation	51

LIST OF ABBREVIATIONS

BIM	Building Information Modeling
CAD	Computer-aided Design
CE	CityEngine
DAE	Digital Asset Exchange
DWG	Drawing
HTML5	Hypertext Markup Language, fifth revision
LoD	Level of Detail
OBJ	3D geometry definition file format
SHP	Esri Shapefile
SKP	SketchUp files
WebGL	Web Graphics Library

1 INTRODUCTION

Visual reconstruction of archaeological data is an interdisciplinary topic that has recently gotten great attention from various scientific fields and industries. Thus this thesis deals with many disciplines, starting with archaeology, cultural heritage and common principles of its visualization, technical approaches of 3D modeling, mathematical discipline of fuzzy set theory handling the calculation of uncertainty, and ending with the suitable way of communicating the results to the society.

This chapter is structured to four main sections which introduce the essential background information from various disciplines that are necessary for completion of this master thesis. The first part presents cultural heritage and its visualisation. Next, it defines the main principles about handling, creating and maintaining 3D models of the cultural heritage, especially visualisation of archaeological features. Depending on the main visualisation principles, this section discusses the tension between veridical and photo-realistic modeling. These two terms are explained later in subsection 1.1.3. Secondly, there is a section devoted to the description of different approaches of 3D modeling. It includes three particular methods: computer-aided design (CAD), building information modeling (BIM), and procedural modeling. Each of the mentioned technologies is explained and the main advantages and disadvantages are highlighted. The next section describes in general mathematical theory called fuzzy logic. Furthermore this part presents possible approach of implementing fuzzy logic into the virtual reconstruction of archaeological sites and solve the the discussed issue about handling uncertainty of archaeological virtual reality. A section about the main motivation and challenges of this master thesis is located at the end of this introductory chapter.

1.1 *Visualisation of cultural heritage*

Cultural heritage can be defined as *“the entire corpus of material signs - either artistic or symbolic - handed on by the past to each culture and, therefore, to the whole of humankind”* (Jokilehto, 2005, p.4). According to Frischer et al. (2000), visualisation of cultural heritage may be also referred to cultural virtual reality whose development started in the 1990s and since then has spread really fast. They indicated that the speed of the popularity of modelling virtual reality corresponds to the evolution of the computer's power. There are many reasons why scientists and even the public are interested in the computer models of cultural heritage sites (Frischer et al., 2000). First of all, Frischer et al. claim, it is very natural since almost everybody is nowadays familiar with the computer environment. Secondly, the digital reconstructions of archaeological sites, buildings or entire cities are pleasant to human eyes and we can get more information and imagination about the heritage in shorter time because visuals are able to transfer the information faster than loads of text files or tables (Denard, 2012).

However it is important to implement technical, aesthetic, and scientific standards into the process of creating the digital reconstructions (Frischer et al., 2000). To solve this, there are two essential documents trying to set a general rules for creating, publishing and maintaining the digital cultural heritage (Denard, 2012). They are named London Charter and Seville Charter and their content is discussed in more detail in following subsections 1.1.1 and 1.1.2.

1.1.1 *London Charter*

London Charter was established in 2006 to ensure the methodological process of computer based visualisations and to provide means of researching and communicating of cultural heritage. It is

currently one of the most advanced international documents in this field (Lopez-Menchero et al., 2011) and it presents the main principles and needs regarding the creation or evaluation of digital reconstructions. The authors of London Charter were successful in dissemination of their work and London Charter has been recently recognized as the benchmark for the heritage visualization processes and outputs (Denard, 2012). Another success that is worth to mention is origination of a new charter, the Seville Charter, which proposes specific guidelines for archaeological community.

Among the main objectives of the London Charter, a few ambitions regarding the computer-based visualizations can be highlighted: the final visualizations should be properly understood and evaluated by the audience, and they should contribute to the further study and interpretation of cultural heritage sites (London Charter Initiative, 2009). Additionally, all the archaeological sources and information used for the modelling of the virtual reality should be appropriately cited and the necessary explanatory text has to be provided. This is described by the primary keynote which states that *“visualisation should accurately convey to users the status of the knowledge that they represent, such as distinctions between evidence and hypotheses, and between different levels of probability”* (London Charter Initiative, 2009, p.2).

London Charter consists of six principles: implementation, aims and methods, research sources, documentation, sustainability, and access. For instance, the “principle of aims and methods” claims that the advantage of each method of visualization (veridical or photo-realistic, schematic or impressionistic, representation of hypotheses or only of the available evidence, dynamic or static) should be considered ahead, taking in account the user’s needs (London Charter Initiative, 2009). Furthermore “documentation principle” declares that *“visualisation outcomes should be disseminated in such a way that the nature and importance of significant, hypothetical dependency relationships between elements can be clearly identified by users and the reasoning underlying such hypotheses understood”* (London Charter Initiative, 2009, p.9). Dependent relationship between two properties means that change of one property will cause a change in the dependent property; for instance, height of the window will generate associated change in the height of the window frame (Denard, 2012). Under these circumstances, Denard (2012) stated that, visualization can serve as an empirical instrument to study the unknown dependencies and hypotheses.

1.1.2 Seville Charter

The above mentioned London Charter takes cultural heritage as a concept, hence there is a need for more specific principles that would determined its constituent parts. Accordingly, the reason why Seville charter has been formed was to provide enough standards for archaeology, specifically in the concept of Virtual Archaeology (Lopez-Menchero et al., 2011). *“Virtual archaeology is scientific discipline that seeks to research and develop ways of using computer- based visualisation for the comprehensive management of archaeological heritage”* (Lopez-Menchero et al., 2011, p.2). Archaeological heritage includes everything that is movable, immovable, extracted, or not yet extracted, and everything that is situated on the surface, underground, or in water, and it can be primary studied using archaeological research methods (excavation or surveying) (Lopez-Menchero et al., 2011).

According to Lopez-Menchero et al. (2011), there are three important terms related to the virtual archaeology whose meanings should be clarified: virtual restoration, virtual reconstruction, and virtual recreation. Firstly, they defined that virtual restoration uses virtual model to visually recreate something that existed in past from the available remaining material (creating only the ruins). On

the other hand, virtual reconstruction visually recovers a building or an object at a given moment in the past from the available evidence and studies carried out by archaeologists and other experts. And finally, virtual recreation consists of the virtual model enhanced by cultural significance, like material culture, environment, landscape, and customs (Lopez-Menchero et al., 2011). This thesis deals with visual reconstruction.

Lopez-Menchero et al. (2011) are authors of the Seville Charter which adopts all the objectives approved by the London Charter and furthermore it adds several new objectives. The most relevant are: establishing guidelines that will give public a greater understanding of archaeology, and defining principles and criteria for measuring the quality of projects. In general, Seville Charter consists of eight principles: interdisciplinary, purpose, complementarity, authenticity, historical rigour, efficiency, scientific transparency, and training and evaluation. One of the most significant principles for this thesis is authenticity; the visualizations are made in a way of how we believe they were in the past (Lopez-Menchero et al., 2011). Therefore it should be always clearly differentiated between: *“remains that have been conserved “in situ”; remains that have been returned to their original position; areas that have been partially or completely rebuilt on original remains; and finally, areas that have been restored or reconstructed virtually”* (Lopez-Menchero et al., 2011, p.4). Last but not least, the principle of historical rigour states about depicting different historical phases; *“a rigorous approach would not be one that shows only the time of splendour of reconstructed or recreated archaeological remains but rather one that shows all the phases, including periods of decline. Nor should it display an idyllic image of the past with seemingly newly- constructed buildings”* (Lopez-Menchero et al., 2011, p.4). Since suitable and correct rendering of a final model is essential for this master thesis, next subsection 1.1.3 analyses this problem in more detail . Finally, Seville Charter asks for a professional interdisciplinary communication and cooperation among the experts from different fields (Lopez-Menchero et al., 2011).

1.1.3 Tension between veridical and photo-realistic modelling

As it is mentioned in Seville Charter, the final visual rendering of a virtual model could be critical so we should obey all the recommendation specified above (Lopez-Menchero et al., 2011). Visual models play a prominent role in shaping public impressions of the past even when the models are inexact (Denard, 2012). Denard (2012) notifies that this convincing power of visualisations can turn out to be a big disadvantage, especially when photo-realism is used, then they can unwillingly lead to an inexcusable perception of historical accuracy. Thus also veridical modeling should be highlighted. Veridical modeling can be referred as accurate or authentic modeling when the audience is not confused with the realistically looking model and they are able to distinguish between fact and fiction related to the background model (Denard, 2012).

However 3D models of archaeological sites are becoming more popular not only for edutainment purposes but they serve as a platform for scientific research about the reconstruction hypotheses (Haegler et al., 2009). Haegler et al. (2009) highlight that the realistically appearing models may confuse even the cultural heritage experts because they present a false impression of certainty of the knowledge about the past. Because even the archaeologists sometimes do not know the exact proportions of the objects and that is the origin of the question of dealing and communicating the uncertainty.

Haegler et al. (2009) propose a solution for the issue of different visualization rendering and uncertainty. They admit that, despite all the scientific rigour commands about clear communication of the uncertainty, public still demands spectacular visual experiences. The solution is not to create

one model, but multiple models. This would naturally solve the expression of uncertainty while the users would be still interested. However this solution is only valid when the creation of multiple models would be efficient (Haegler et al., 2009). Consequently, more details about the uncertainty are analysed in section 1.3.1 discussing fuzzy logic approach in archaeological virtual reconstruction and section 3.4 about handling uncertainty and reliability of the 3D models.

1.2 *Possible approaches of modelling*

There are many technologies for creating 3D models and to characterize all of them would be out of the scope of this master thesis. Thus, in the following subsections there are descriptions of three recently discussed approaches. Firstly, the traditional Computer Aided-Design (CAD) technique is characterized. Secondly, the newest approach of Building Information Modeling (BIM) is presented. And lastly, procedural modelling is defined and it is clarified why it has become so popular in visualisations of cultural heritage.

1.2.1 *Computer Aided-Design (CAD)*

Traditional approach computer-aided design (CAD) is used for design and design documentation. CAD software allows designers to change manual drafting to an automated process on a computer screen (Minialoff, 2000). The original purpose of CAD was to display 2D geometry by graphical elements, such as lines, arcs or symbols (Howell et al., 2005). Soon, it was used to model 3D geometries because of the simplicity, speed of modeling, its visualization possibilities, realistic rendering, and lightning effects (Howell et al., 2005). Regarding Bilalis (2000), the advantages of CAD are: accurate generation and modification of a final object, creation of design analysis in relatively short time, or usage of a solid and surface modeling to provide an unambiguous 3D representation. Additionally, 3D models can be exported to other platforms in several formats and they can be used as a the communication medium between groups of people from various research fields (Bilalis, 2000).

However, several disadvantages appear when there is a need for the modeling of extensive buildings with several parameters or even entire cities with street network and vegetation. In these cases CAD seems to be time consuming and ineffective, thus it is too expensive for the research fields like archaeology (Müller et al., 2006b). But in spite of these disadvantages, CAD modeling is still necessary part of 3D modeling. Because even when the procedural modeling is used there is a need for terminate shapes that are mainly presented by 3D objects created in CAD environment.

1.2.2 *Building Information Modeling (BIM)*

One of the recently published approaches is building information modeling (BIM) which was introduced in 2003 (Howell et al., 2005) and since then it has attained extensive attention in architectural, engineering, and construction industry (Azhar, 2011). BIM, sometimes called intelligent design information, was designed for the three-dimensional modelling of buildings. According to Ernstrom et al. (2006), this technology mainly focuses on the construction industry and it brings a new approach of model-based process that replaces the common usage of independent 2D blueprints by one virtual model of the whole building rendered in 3D (Ernstrom et al., 2006). This one particular model is helpful during all the stages of construction because it covers all information about planning, design,

construction itself and operation of a facility, for instance it includes information about air condition unit or heat distribution (Azhar, 2011). Moreover, there is a possibility to visualize the constructing site in 4D (adding time) or 5D (adding quantities and cost of the materials) (Ernstrom et al., 2006).

Basically, BIM is one of the latest generations of object-oriented CAD systems (OOCAD) where the 2D symbols are replaced by 3D building elements that can be displayed in multiple views and they are able to carry even non-graphic attributes (Howell et al., 2005). In my point of view, despite the huge benefits mentioned above, using BIM approach is more suitable for the construction field, where there is a need to simulate the complex operation of a facility, than virtual heritage visualisation where mostly only the exterior parts and design issues have been dealt with so far.

1.2.3 Procedural modeling

Procedural modeling offers a good alternative to both of the above discussed approaches. It limits tedious manual modeling and it is able to deal with inflexible nature of static geometrical site models, thus handling the uncertainty (Haegler et al., 2009). It presents a cost-effective solution, because other approaches of 3D modeling are often too expensive (Watson et al., 2008). This is especially welcomed in the domain of cultural heritage visualization where the budgets are often small (Noghani et al., 2012). The main potential of procedural modeling lies in the visualizations of entire city areas or testing several hypotheses; moreover it can be applied in research, education, urban planning, creation of virtual environment and simulations, and even in the entertainment market (Parish et al., 2001). Parish et al. (2001) proposed an idea to connect procedural modeling with the shape grammars that are iteratively evolving the object's design by creating more and more details.

Müller et al. (2006b) have recently presented in their work a new shape grammar for procedural modeling called CGA shape (computer graphic architecture shape) based on L-systems. They claim that CGA shape is suitable for modeling cities with high quality design and geometric detail. Firstly, a simple mass model is assigned to a building, representing the highest level of the hierarchical structure. Then, the building is subsequently structured to lower levels (walls and roof), and finally the details of the lowest levels (windows and doors) are added (Müller et al., 2006b).

Another implementation of procedural modeling was demonstrated by Wonka et al. (2003) when they proposed a new type of design grammar called "*split grammar*", a solution for generation of geometric details on façades. Moreover, especially in archaeology, procedural modeling offers an opportunity to update the model more often as the scientific knowledge increases over time of excavations and further analysis (Haegler et al., 2009). However Haegler et al. (2009) highlight few disadvantages, one disadvantage of this technique is that certain programming knowledge has to be gained in order to create the rules and generate the model. Another drawback is that procedural modeling better suits regular shaped buildings and thus modeling of irregular and complex shapes might be too complicated (Haegler et al., 2009). Further characteristics of procedural modeling and its usage in archaeology is discussed in section 3.3.

1.3 Fuzzy logic

Fuzzy logic is a branch of mathematics based on fuzzy set theory that is an extension of set theory. Fuzzy set theory was firstly introduced by Zadeh in 1965 and it proposes a strict mathematical

framework for handling classes of objects with imprecisely defined criteria or memberships (Zadeh, 1965). Dealing with these imprecisely defined classes is essential for human thinking because humans are used to breaking down the whole world into categories (Zimmermann, 1991). However these categories often have blurred edges and uncertain membership (Smithson et al., 2006). The fuzzy theory is applicable among many fields of science because its purpose is to naturally and systematically deal with unsharply defined criteria or vagueness (Zimmermann, 1991). Thus, it became a very powerful modeling language that is able to process large fraction of real-life uncertainties (Zadeh, 1965).

The foundation of fuzzy set theory is set theory which is described as “*a mathematical calculus for dealing with collections of objects and certain relationships among these objects*” (Smithson et al., 2006, p.4). Smithson et al. (2006) provides following definition of sets and their operation. Sets are usually connected with a rule determining the memberships inside of it. This rule is of the utmost importance since it connects the objects to each other and thus it must be explicitly defined. There are four common operations that can be applied on sets: union, intersection, negation, and inclusion. But it can happen that it is not possible to define the rule precisely, thus there is a need of a fuzzy set since classical sets are not able to deal with uncertainty within their rules (Smithson et al., 2006).

A fuzzy set is defined as “*a class with a continuum of grades of membership*” (Zadeh, 1965, p.339) and it handles mostly one particular kind of uncertainty, named “degree vagueness” (Smithson et al., 2006). Degree vagueness uncertainty appears when there is property that can be assigned to objects with varying degree of membership (Smithson et al., 2006). So Zadeh (1965) characterized a fuzzy set by a membership function m_A that varies between $[0, 1]$ dependent on the grade of membership. The higher the grade of membership, the higher the degree that the particular object is a member of the particular set (Zadeh, 1965). Accordingly, a fuzzy truth function can be defined as varying between 0 (false) and 1 (true) assuming also intermediate values - “uncertain” (Niccolucci et al., 2004). At this time it is necessary to highlight the differences between probability theory and fuzzy set theory. Unlike probability theory, degrees of membership of all objects do not have to sum to 1 in fuzzy set theory, which means that any number of objects can have high value of the degree of membership (Smithson et al., 2006). However, an object’s memberships to several classes must still sum to 1. As set theory, fuzzy set theory includes the same four basic operations but in addition it includes other operations such as the modifiers concentration and dilation, and the connective fuzzy aggregation (Smithson et al., 2006).

1.3.1 Fuzzy logic approach in archaeological virtual reconstruction

As it was mentioned in the previous sections, the existence of uncertainty and the need of dealing with it is becoming a crucial issue in the modeling and visualizing of cultural heritage (Haegler et al., 2009). Fuzzy logic appears as a suitable solution since it is able to deal with uncertainty and imprecision, and it can be applied in many different areas of research and industry (Niccolucci et al., 2004). The recent studies using different modifications of this method are described in section 2.2. However, in conclusion, it does not matter which method is used, but the results have to be clearly communicated to every user so they would be able to evaluate the creation process and make their own opinions (Niccolucci et al., 2004). Implementation of these approaches may change the characteristics of virtual reconstructions. Archaeological visualizations with fuzzy set theory do not have to be used only for presenting the perfectly looking virtual model as a “*objective truth*” but they may provide space for further scientific analysis, hypotheses, and expressing of the data transparency (Niccolucci et al., 2004). These methods seems to be perfect for the purposes of current modeling of cultural heritage when there is a tremendous emphasis given by the archaeological field on the visualization of

uncertainty and expressing of the reliability of the data used for the reconstructions (Denard, 2012; London Charter Initiative, 2009; Lopez-Menchero et al., 2011).

1.4 *Motivation, Challenges*

The motivation of this thesis is to create a three-dimensional virtual model from archaeological data of the Sanctuary of Diana at Nemi, in Italy. Archaeological data used in this master thesis were provided by archaeologist Dr. Francesca Diosono and by the department of Cartography at Technische Universität München (TUM) which takes part in the research group at the excavation site in Nemi. In 2013 a 3D model of the excavated ruins of the sanctuary was generated but an entire 3D reconstruction has never been done, hence this is the first attempt to create a virtual model of this sanctuary. The model is created using procedural modeling in the environment of CityEngine software.

Several challenges appear in the scope of this master thesis. The first is the generation of the 3D model using procedural modeling techniques and prospering of its main benefits related to the possibility of testing hypotheses or implementing knowledge of newly found evidence. The second challenge is handling and visualizing the uncertainty of archaeological data since some of the parameters of the final building of sanctuary are not precisely defined and even the archaeologists are not able to determine the exact value. According to the literature (Hermon et al., 2002, Hermon et al., 2003) reliability and uncertainty of archaeological data can be solved by implementing fuzzy set theory. Thus the fuzzy set approach will be implemented to calculate and express the uncertainty in a correct and comprehensive way. Thirdly, this thesis deals with different rendering of the final visualization. More specifically the advantages and disadvantages of photorealism and non-photorealism are examined. This variation in the rendering is related to diverse end-users and the primary purpose of the virtual model. The last challenge is to publish the 3D reconstruction on-line and to ensure that the user-interface will be consistent with the principles of the London Charter and the Seville Charter (London Charter Initiative, 2009; Lopez-Menchero et al., 2011), ie. all the used sources and documentations should be communicated, the metadata should be available, and the reliability of the virtual model should be pleasantly expressed. However, the principles of the Charters should be obeyed in the whole process of the reconstruction.

This chapter provides an overview of the recent studies using the same methods and principles that are compiled within this thesis. Unfortunately, only a few projects deal with procedural modeling and handles uncertainty at the same time. This fact leads to the structure of this chapter which is divided into the three parts subsequently discussing different topics. The first section 2.1 presents studies that are implementing procedural modeling in order to create the 3D reconstruction of archaeological data. The next section 2.2 gives an overview of the approaches of different studies quantifying the uncertainty using fuzzy logic. The third section 2.3 defines possible methods of visualizing the uncertainty which are demonstrated on the results of recent studies.

2.1 *Archaeological reconstruction using procedural modeling*

The amount of projects using procedural modeling, especially in the CityEngine, for the generation of 3D reconstruction of archaeological data has increased recently. Because there are many studies implementing procedural modeling, this section provides a brief description of the main discussed works. First, the extensive project Rome Reborn 2.0 used procedural modeling techniques in order to generate entire city of ancient Rome in 320 AD (Dylla et al., 2008). However hand modeling approaches were also needed to show the high detail resolution. Example of this project is depicted in figure 1.

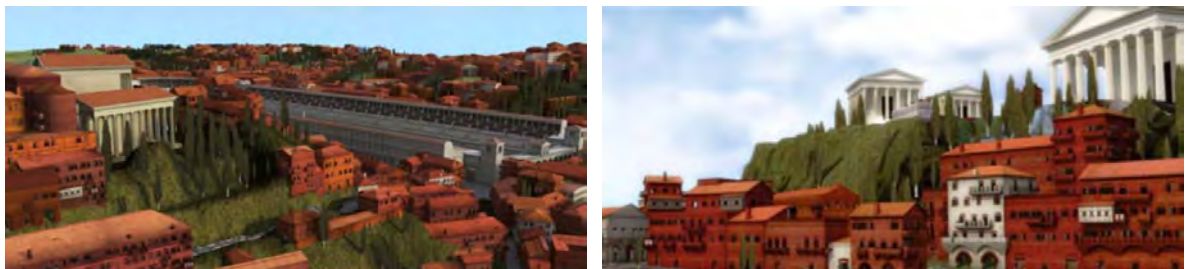


Figure 1: Illustrations of the project Rome Reborn 2.0, source: Dylla et al., 2008

The second project shows the reconstruction of another Roman city Pompeii destroyed by the eruption of the volcano Mount Vesuvius (figure 2). This project was created by Maïm et al. (Maïm et al., 2007) as a test case of the CityEngine. It also determines four different levels of detail for the CityEngine models: LoD 0 presenting the mass model, LoD 1 showing the model in low detail, LoD 2 presenting the standard visualization, and LoD 3 offering high detail reconstruction (ESRI, 2014b). Additionally, Maïm (2007) was interested in populating the city of Pompeii, so he introduced method of showing its inhabitants and triggering their behaviour.

Thirdly, procedural reconstruction of Mayan buildings in Xkipché was completely done using procedural modeling techniques based on the archaeological data; only the complicated ornament areas were created with traditional mesh modeling software and they were inserted into the CityEngine project as assets (Müller et al., 2006a). The authors of this project were proved that procedural modeling is a promising tool for archaeological reconstructions, because it allows precise encoding of data, accurate 3D model generation, parameter based modeling, and fast creation of different types of buildings.



Figure 2: Previews of the 3D reconstruction of the ancient city Pompeii, source: Maïm et al., 2007



Figure 3: Detailed reconstruction of one of the highly ornamented buildings in Xkipché, source: Müller et al., 2006a

Other projects using the advantages of procedural modeling are for instance: creation of classical Roman architecture (Noghani et al., 2012), Roman housing architecture (Müller et al., 2005), or creation of the traditional Balinese settlement (Di Angelo et al., 2013, Ferschin et al., 2013).

2.2 Quantifying the uncertainty using fuzzy logic

There are several methods that use fuzzy logic for calculation of the uncertainty. Two major case studies dealing with the reliability in archaeological virtual reconstructions are: project of Niccolucci et al. (2004), and work of Tepavčević and Stojaković (2013). Hermon (2002) and Niccolucci (2004) implemented fuzzy logic concepts into archaeological classification and they proved that it helps to increase reliability of the archaeological results. Firstly, Niccolucci et al. defined the creation of an archaeological model as a *"stepwise process in which one starts from an initial model M_0 , possibly empty, placed at position x_0 ; at step n a new model M_{n+1} is built from M_n adding a new detail m_{n+1} in an absolute position x_{n+1} "* (Niccolucci et al., 2004, p.2). Position x are vectors that are uniquely determining positions of objects in space. This process can be explained with an example of a temple reconstruction. In the beginning there is only a representation of archaeological remains - this is equivalent to M_0 . Then, corresponding walls are added with attributes of height and material. Addition of the first wall is equal to addition of the first detail m_1 at position x_1 . Next, when the last wall is created we got a new model M_k . Following, other features like windows, doors, columns,

roof, etc. are added. Every step increases the complexity of the model but it reduces the model's reliability (Niccolucci et al., 2004).

To compute the reliability of a virtual model they established a scale for reliability as an interval $[0, 1]$, where 0 is “*totally unreliable*” and 1 is “*absolutely reliable*”. Next, reference has to be given to the model because the model is not reliable by itself, only when it is referred to a specific problem. It is necessary to highlight that this approach does not consider any time frame, thus it is assumed that the model is temporally reliable. The reliability function r can be considered as a fuzzy truth value of models which uses the minimum of the fuzzy truth function f of its operands. The problem of reliability is then split into absolute reliability $r^{(a)}$ taking into account the reliability of the object *per se*, and relative reliability $r^{(r)}$ which considers the compatibility of the object with the context, which presents previously chosen details and the general characteristics of the model. Thirdly, there is also a positional component of reliability $r^{(p)}$ that is dependent on newly added details with respect to the already generated model. The reliability of the final model M can be defined like this:

$$r(M) = \min_{k=1, \dots, n} (r_0, r_k^{(a)}, r_k^{(r)}, r_k^{(p)})$$

where r_0 is the reliability of the initial model and the reliability of each newly added details is divided into its absolute, relative, and positional component. Thus the final reliability equals the lowest reliability of its sequentially added details. However the result depends on the order in which the details are added because the reliability index can vary according to the characteristics of the models created in previous steps (Niccolucci et al., 2004).

On the other hand, Tepavčević and Stojaković (2013) proposed another approach for group of objects enhanced by the theory of probability. Contrary to reliability components used by Hermon and Niccolucci in previous method, Tepavčević and Stojaković defined correction factors describing the grade of objects in the corresponding fuzzy set. Correction factors K characterize assessment of the probabilities to the results attributes; for instance, shape factor K_f , style factor K_s , and their arithmetic mean K_c . Thus the correction factors gain values between 0 and 1. Then, probabilities of occurrence of every detail and parameter of the object consider adequate correction factors. The final result is a table showing a list of probabilities of the single characteristics added in the sequenced predefined steps. In the end, their resulting visualizations proved that fuzzy logic can be nicely combined with procedural modeling and CGA shape grammar.

2.3 Visualizing the uncertainty

There are many methods from different fields of science that were used to visualize uncertainty. There are also many approaches how to classify these methods. According to MacEarchen (1992) these methods of how to visualize the lack of information can be divided into three main categories: Firstly, separate illustrations can be used to visualize the attribute of interest and the appropriate uncertainty which enables comparison of the illustrations (Slocum et al., 2009). Secondly, these two phenomenon can be visualized on the same illustrations using suitable visual variables. Third category presents using of interactive visualizations including data exploration tools or animations. Contrary to MacEarchen, Griethe (2006) and Sifniotis (2012) suggests dividing the visualizing methods into five categories which are described in the following paragraphs. Furthermore, description of the individual categories are accompanied with corresponding illustrations from already published projects.

Free graphical variables

In most cases the visualization of uncertainty is directly mapped into the graphical representation of the 3D model. Unused graphical attributes are used for encoding the uncertainty. The graphical variables can also be referred as an intrinsic visual variables which are intrinsic to the display (Gershon, 1998). Thus, this method affects color, size, position, texture, lighting, shading, clarity, edge crispness, and opacity or transparency values (Griethe et al., 2006, Sifniotis, 2012). Furthermore this method also includes pseudocoloring, *“a technique for representing scalar, interval, or ordered data values by using a sequence of colours”* (Sifniotis, 2012, p.41-42). In her work related to the uncertainty of 3D models of archaeological sites, Sifniotis implemented the approach of pseudocoloring with a combination of transparency (figure 4).

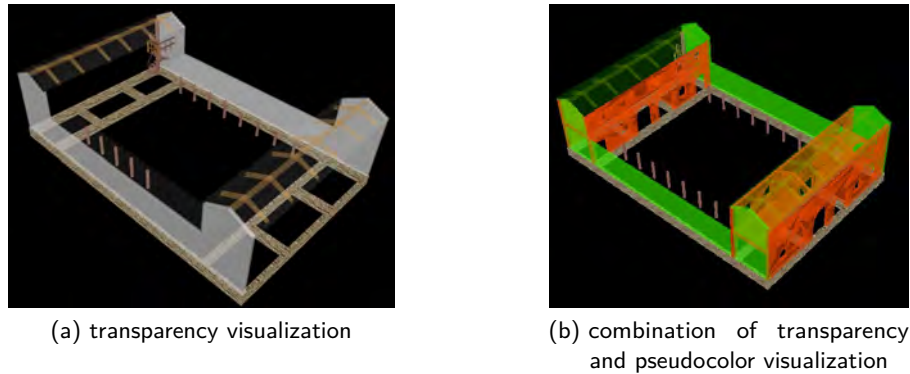


Figure 4: Visualization of the uncertainty using pseudocoloring and transparency, source: Sifniotis, 2012

Additionally, Sifniotis (2012) proposed an example of using different textures as a indicator of the uncertainty. Firstly, she used pseudocoloring and chessboard texture to express the results of her complex uncertainty calculation (figure 5a). Second approach shows the results of pseudocoloring and intensity of noise (figure 5b).

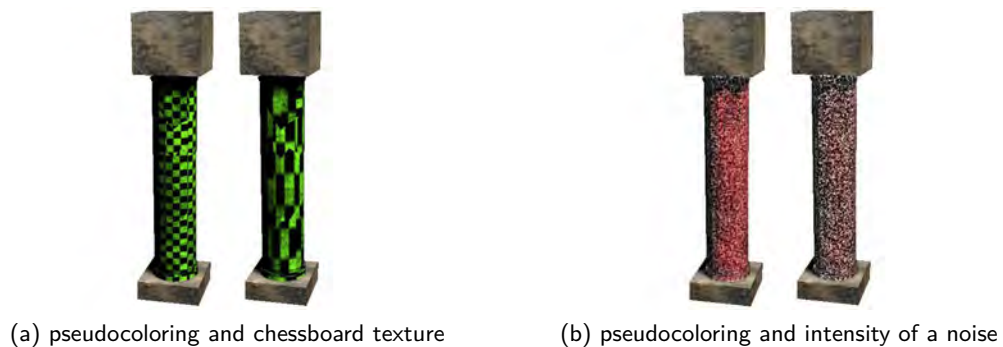
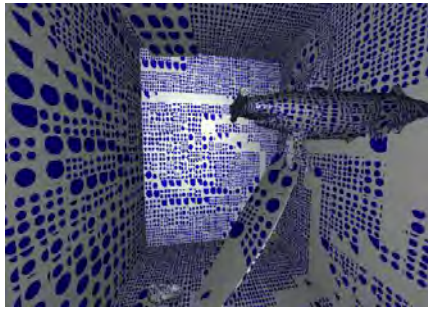
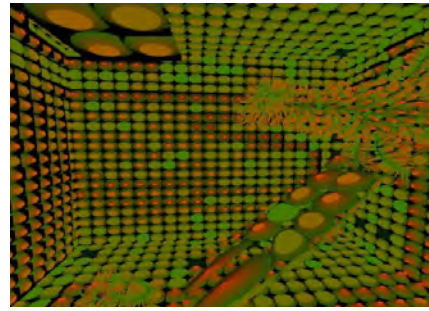


Figure 5: Visualization of the uncertainty using pseudocoloring, chessboard texture, and intensity of a noise, source: Sifniotis, 2012

Textures as a mean of visualizing uncertainty are among the others described by Pang et al. (1997). One of their outputs is mapping an attribute value into the 2D circular textures (figure 6a). Then, they implemented this approach into 3D modeling while mapping the value into the 3D solid textures as depicted in figure 6b.



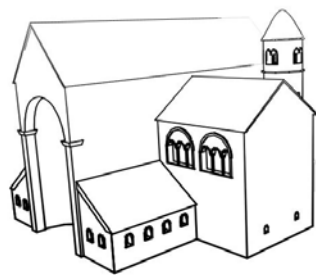
(a) 2D circular textures



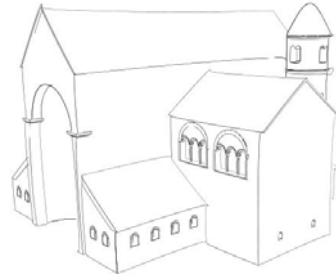
(b) 3D solid textures

Figure 6: Mapping an attribute value into the 2D circular textures and 3D solid textures, source: Pang et al., 1997

Strothotte et al. (1999) implemented different edge rendering in their work in order to visualize the uncertainty. Figure 7a shows the visualization using different variations of edge saturation where the upper parts of the building are less certain. Secondly, they used variation of edge sketchiness presented in figure 7b.



(a) edge saturation



(b) edge sketchiness

Figure 7: Visualizing uncertainty by variation of edge saturation and edge sketchiness, source: Strothotte et al., 1999

Side by side

The side by side method compares different visualizations of data. It could be, for example, a comparison of certain and uncertain models (Sifniotis, 2012) or multiple realistically looking models representing different hypotheses (Haegler et al., 2009). One of this approach using procedural modeling was implemented by Tepavčević and Stojaković (2013) whose results of side by side comparison are demonstrated in figure 8.

The benefit of procedural modeling, which lies in the fast rendering of different types of a model, is directly highlighted in the report of the consortium called 3D-COFORM (Van Gool et al., 2013). The main aim of this consortium is to “*establish 3D documentation as an affordable, practical and effective mechanism for long term documentation of tangible cultural heritage*” (3D-COFORM, 2014). 3D-COFORM suggests using CityEngine and procedural modeling in order to create many side by side visualizations which presents the uncertainty knowledge of the final building's appearance shown in figure 9.

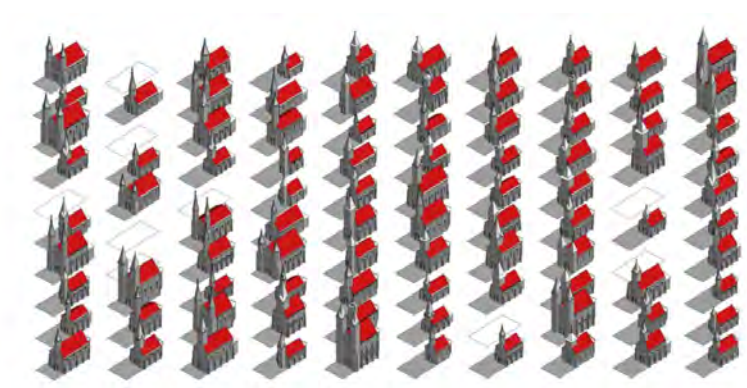


Figure 8: Visualizations of different results of the archaeological reconstruction of the chapel in Vojvodina, source: Tepavčević et al., 2013



Figure 9: Side by side illustration of the uncertainty modeling generated in CityEngine, source: Van Gool et al., 2013

Animation

Third visualization technique exploits the benefits of dynamic representation including parameters like speed, blinking, duration, and motion blur (Griethe et al., 2006). Animation can include changing different results within the scene. Additionally, as Zuk et al. (2005) introduced, animation can express the temporal uncertainty as depicts figure 10. Individual frames of the animation shows data from different parts of the timeline.

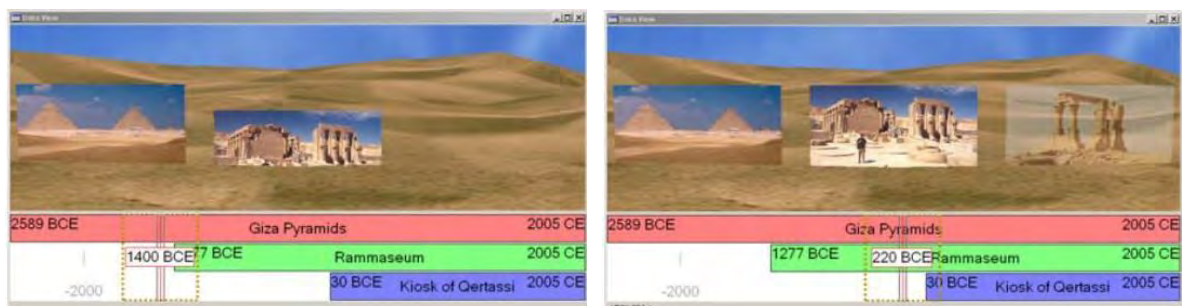


Figure 10: Visualizing temporal uncertainty using animation, source: Zuk et al., 2005

Interactive representation

Uncertainty could be expressed by user interaction, the best example would be click-able objects offering uncertainty information after mouse interaction (Griethe et al., 2006; MacEachren et al., 2005). One of these projects that offer this interactivity is for instance work of Van der Wel et al. (1998).

Integration of objects

The last method considers implementation of external objects to the 3D model, like glyphs, error bars, and text labels (Sifniotis, 2012). These external objects are also called extrinsic visual variables since they are additionally added to the display (Gershon, 1998). Sifniotis has shown the possible approach of implementing glyphs into the 3D models as illustrated in figure 11. However usage of the glyphs can be visually overwhelming (MacEachren et al., 2005).

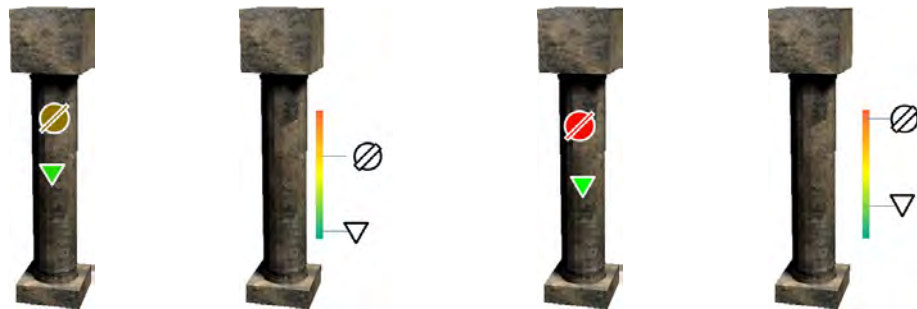


Figure 11: Visualization of the uncertainty using extrinsic visual variables, source: Sifniotis, 2012

Following figures depict another solution using glyphs. These figures present work of Pang et al. (1997) who used for instance spherical glyphs which varying sizes express desired attribute (figure 12a). Next, Pang encoded uncertainty to the vector glyphs visualized in figure 12b. Lastly, they implemented line glyphs which lengths correspond to attribute value, for example uncertainty value (figure 12c).

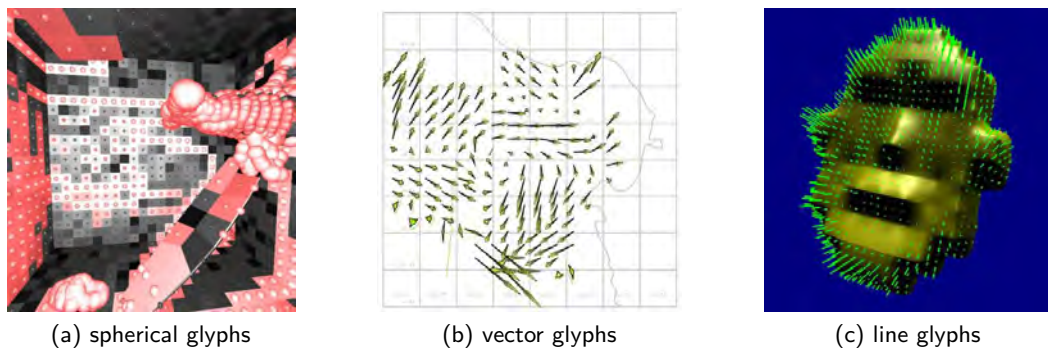


Figure 12: Visualizing an attribute value using spherical, vector, and line glyphs, source: Pang et al., 1997

3 METHODOLOGY

This chapter deals with the methodological part of this thesis. First of all, section 3.1 covers the description of the archaeological site in Nemi. Next, section 3.2 provides an overview of available archaeological information such as a brief introduction to Roman architecture of temples, characterization of technical data, and actual knowledge about the Sanctuary of Diana. Thirdly, section 3.3 explains the procedural modeling technique and its usage in archaeology, it gives a brief introduction about the CityEngine tool and describes the workflow for visualization of cultural heritage. The last section 3.4 introduces the method of dealing with uncertainty which use fuzzy logic. Particularly it defines techniques of quantifying the uncertainty followed by the description of possible methods of uncertainty visualization.

3.1 Description of archaeological site in Nemi

The archaeological site investigated in this thesis is situated in the municipality of Nemi in central Italy. Nemi is located 30km south-east from Rome (figure 13a) and it belongs to the province of Rome. Figure 13b shows the exact location of the excavations in the northern part of the Nemi basin. This basin is a former volcanic crater and its southern part is covered by the lake having the same name Nemi. Since 1984, this area is part of the national park called Parco Naturale dei Castelli Romani (Storemyr, 2004).

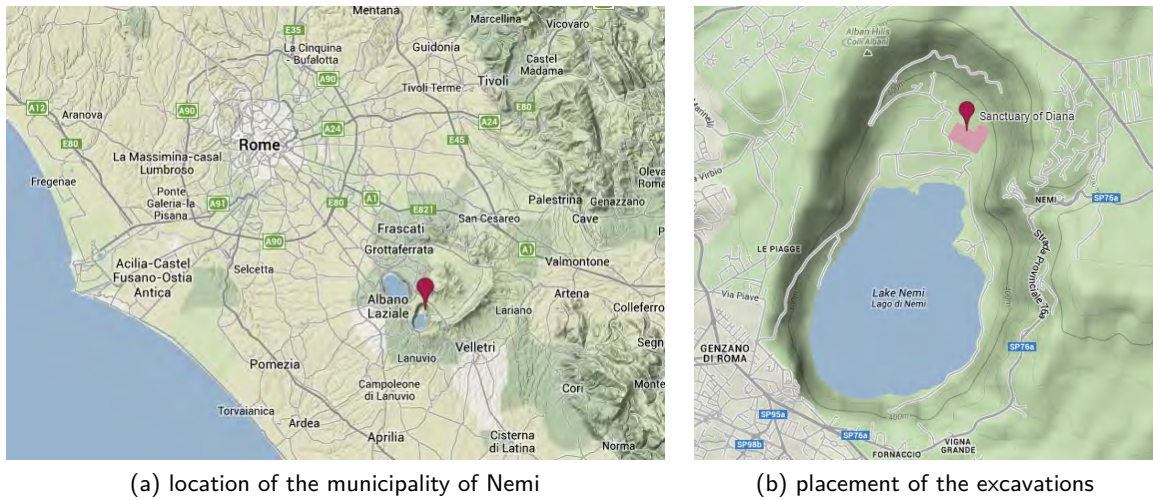


Figure 13: Location of the excavations of the Sanctuary of Diana at Nemi, Italy, source: GoogleMaps, 2014

Thanks to the fertile soils human habitation can be tracked far into prehistory yet the most famous period belongs to Roman times (Storemyr, 2004). Moreover, two important medieval cities are built around this basin: Nemi and Genzano di Roma. Many Roman structures have been discovered within this basin, for instance: villas, cisterns, assumed aqueducts, roads paved with basalt slabs, etc (Peters et al., 2012; Storemyr, 2004). The area of ongoing excavation is extensive as illustrated in figure 16 in section 3.2.2.

This thesis focuses on the ruins of ancient temple named Sanctuary of Diana. Diana Nemorensis, sometimes also known as Diana of Nemi or Diana of the Wood, was an Italian goddess worshipped in

ancient Roman religion (Peters et al., 2012). According to Roman mythology, Diana was the goddess of the hunt, moon, and birthing; moreover she has been usually associated with wild animals and woodland (Nemi to Nottingham, 2014). This temple was connected by the *Via Virbia* to one of the most important roads in Rome called *Via Appia* which connected Rome and the current Brindisi (Peters et al., 2012). The only remains of the Sanctuary of Diana are in the form of excavated ruins that are illustrated in figure 14.



Figure 14: Photograph of the excavations of the Sanctuary of Diana, taken in 2013, source: Francesca Diosono, Department of Cartography at TUM

The reason why the archaeological site of Nemi was selected for this thesis is that the Technische Universität München (TUM) has been cooperating with the archaeologist group studying this area more than three years. Thus many studies have been carried out and there should be enough available data. Nevertheless this is the first attempt to create a virtual reconstruction of the Sanctuary of Diana using procedural modeling.

3.2 Available archaeological knowledge

Sources of archaeological information related to the Temple of Diana can be structured into three parts. Firstly, literature sources were analysed in order to get familiar with the topic of Roman temple architecture. Subsection 3.2.1 provides deeper insight to this knowledge. Secondly, there are technical datasets containing different file formats such as shapefiles (SHP) or CAD formats (DWG). This type of information, which was created by a research group from TUM, is discussed further in subsection 3.2.2. The third source of knowledge covers the materials from archaeologists in the form of pictures, sketches or CAD files. However some parts are less tangible since they were gained from oral or email conversation directly with the archaeologists, especially Francesca Diosono. The need for this type of information, which is described in the subsection 3.2.3, arose mainly during the 3D model generation and modeling of uncertainty.

3.2.1 General description of Roman temple architecture

One of the most extensive literature sources about Roman architecture is a piece from Roman architect Vitruvius Pollio (1914) called *“De architectura”* or *“The Ten Books on Architecture”*. This book is the

oldest treatise (ca. 30 BC) on architecture that has survived into the modern era. The reason is that Vitruvius had a unique and innovative way of writing. He used discursive style in the construction of technical texts and he presents architecture in a new rhetorical format explicitly determined for the public domain (Patterson, 1997). Vitruvius aimed to cover construction of the most significant Roman buildings accompanied by schematic and detailed illustrations. One of these illustrations is presented in figure 15 that shows dimensions of the individual parts of the Ionic column. The buildings are defined within the terms of actual measurements and proportions; which are related to the proportion of the human body (Vitruvius Polio, 1914). The measurements are in Greek Feet which is approximately 29,6 cm (Duncan-Jones, 1980). Regarding the topic of this thesis, the most essential parts of Vitruvius's work are Book III and Book IV which are devoted to the construction of temples.

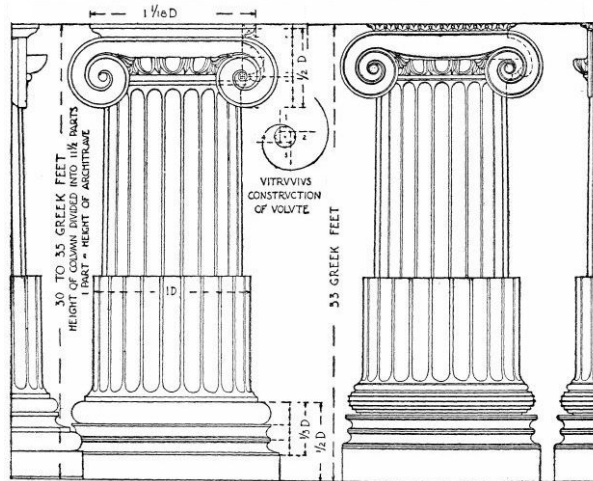


Figure 15: Vitruvius's illustration of the dimension of the Ionic column, source: Vitruvius Polio, 1914

Furthermore, Vitruvius mentions Temple of Diana at the end of his fourth book: *"There are also other kinds of temples, constructed in the same symmetrical proportions and yet with a different kind of plan: for example . . . the temple of Diana in her sacred grove, with columns added on the right and left at the flanks of the pronaos"* (Vitruvius Polio, 1914, p.125). This may seem as a huge advantage for further work. However his indication means that the plan of this temple is different to all other plans he was describing. Fortunately proportions of this temple are typical: *The length of its "cellae is twice the width, as in other temples; but all that we regularly find in the fronts of others is in these transferred to the sides"* (Vitruvius Polio, 1914, p.125).

When talking about the construction of the temple, it is an advantage to analyze each main part consecutively. Omitting the general layout/plan of columns and walls since the Temple of Diana is beyond the common definition, we first distinguish between five styles of temples regarding the proportions of intercommunications and columns. These styles are: pycnostyle, systyle, diastyle, araeostyle, and eustyle. Then, there is an interesting reference to structure of steps: *"the steps in front must be arranged so that there shall always be an odd number of them; for thus the right foot, with which one mounts the first step, will also be the first to reach the level of the temple itself"* (Vitruvius Polio, 1914, p.89). When talking about proportions of the base, capitals, entablature, and structure of ornaments, it must be always distinguished between Ionic, Doric, and Corinthian order. Next, proportion of doorways depends on Doric, Ionic, and Attic style. Finally, altars should be oriented towards east and they should be placed on a lower level than all the statues located in the temple (Vitruvius Polio, 1914).

3.2.2 Technical data about excavation in Nemi

Thanks to the cooperation between the TUM research group and the group of archaeologists, there is lot of accessible data related to the sanctuary of Diana. Among others, CAD files, shapefiles, sketches and photos are available. One of the sketches is presented in following figure 16 and it shows the extensive area of ongoing excavations. Sanctuary of Diana is represented by the complex structure approximately in the middle of the figure and it is one of the most explored part of this excavation.

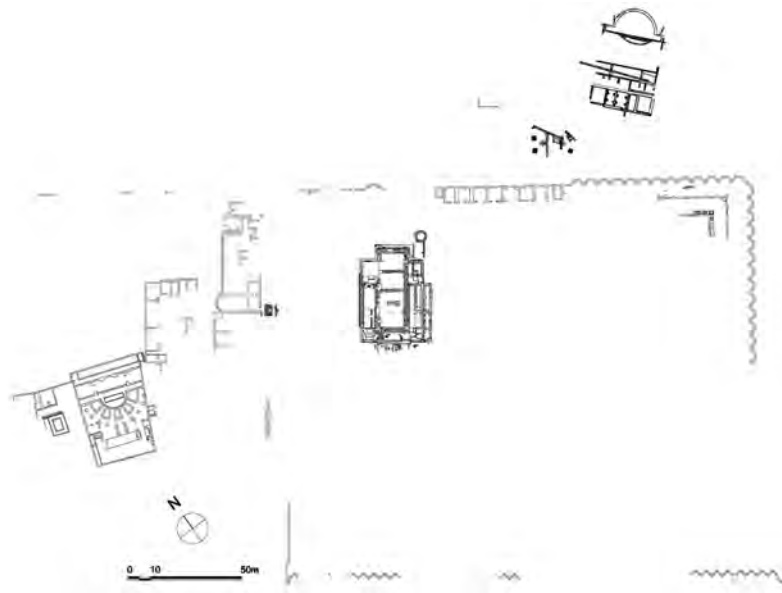


Figure 16: Sketch of the excavation area situated in Nemi, Italy, source: Francesca Diosono, Department of Cartography at TUM

The essential parts of the technical data needed for the precise 3D reconstruction of the temple were the detailed SHP and CAD files of the Sanctuary of Diana containing every single piece of stone which has been found. A print-screen of one of these shapefiles, which was mainly used for measuring the real dimensions of the individual parts of the temple, is shown in figure 17.

Unfortunately, this type of shapefile is impractical for the purpose of 3D model reconstruction in the CityEngine (more about this tool is in subsection 3.3.1), because of its high detail. Thus a generalized shape containing only the rectangular outline has been used and divisions to rooms are subsequently done by CGA rules in the CityEngine. Nevertheless not using such a detailed dataset might be unpleasant and that's why this shapefile is included in the resulting 3D scene. More information about the final rendering is written in section 4.3.

3.2.3 Information about the Sanctuary of Diana

As indicated above, this section has two parts: it begins with the archaeological sources in the form of files and it ends with the description of the less tangible archaeological information gained from conversation with the archaeologists. Important information was gained from the CAD files containing sketches of the excavated pieces of the temple; for instance structure of column shafts, or shapes of entablature remains. This knowledge was mainly used for modeling of terminate shapes described more in subsection 4.1.2 on page 33. Other data determines that the Sanctuary of Diana was sequentially constructed in three time periods. The division of the reconstruction sequences to

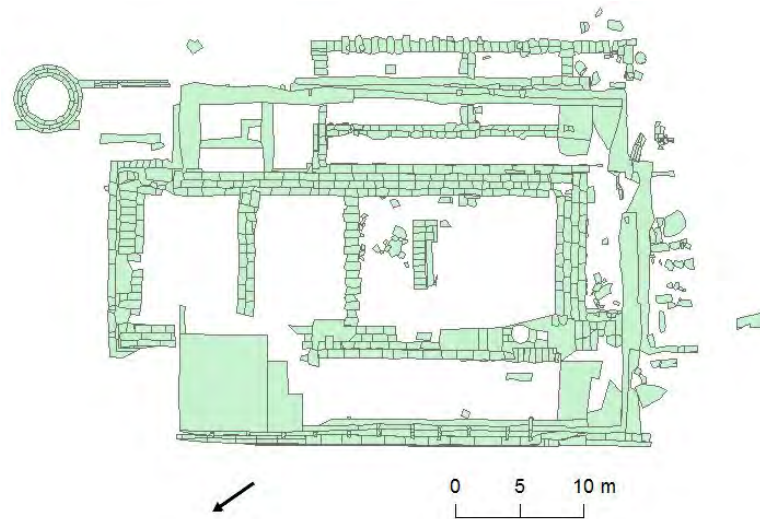


Figure 17: Detailed sketch of the excavations of the Sanctuary of Diana at Nemi, Italy, source: Department of Cartography at TUM

these three phases is depicted in figure 18. However it was decided that only the third phase will be reconstructed within the scope of this thesis and thus figure 18c has become the starting point for the virtual model.

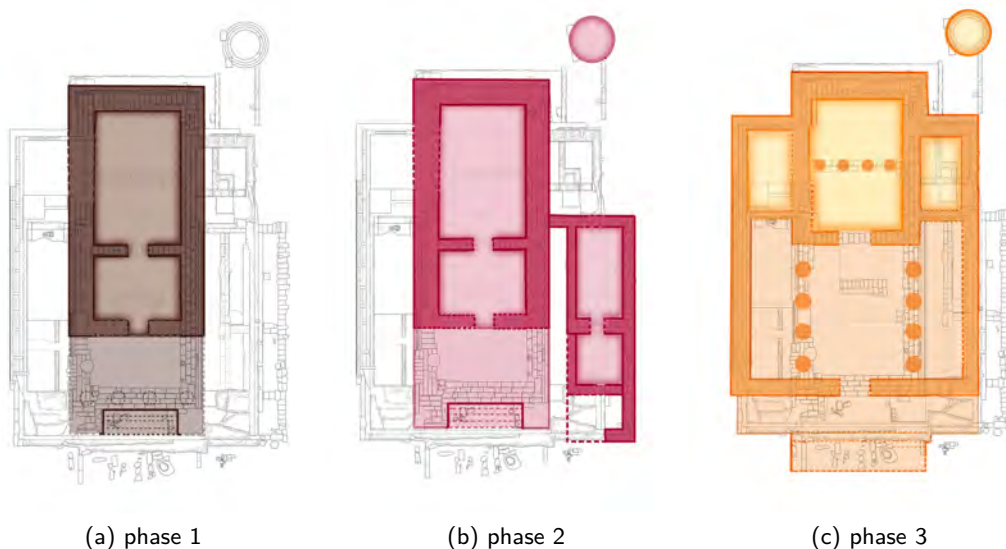


Figure 18: Depiction of the different construction phases of the Sanctuary of Diana, source: Francesca Diosono

The second part of this subsection deals with information gained by oral and email conversation with archaeologist Francesca Diosono. A critical piece of information was the result of the recent archaeological research (spring 2014) according to which the front wall visible on the figure 18c was more likely a colonnade than a wall. Furthermore, the main topic of the discussions was about deeper characteristics of individual parts of the temple, for instance:

- roof tiles were made of bronze, so were the pediment statues

- the circular shape located next to the temple (figure 18) was a podium for a statue that was donated to the temple during the second phase
- there was a window with glass window panes in one of the temple walls
- there was a statue of Diana in the main cella, but its exact shape is not known since only its head was found
- there was a mosaic on the podium floor and in the cella, part was probably simple white mosaic and part had a black and white pattern - but the distribution is not precisely known
- the capitals of columns were Corinthian, however there is no knowledge about their bases
- there was a pavement surrounding the temple consisting of regularly compiled small stones

Last but not least, archaeological knowledge and experiences were needed to define the uncertain places of the temple and quantify their level of reliability. In order to ensure uniformity and quality of the data, first a table containing the temple's objects was created and second, archaeologist filled in the missing values related to the reliability of each object, or varying reliability of different object's variations, and textual explanation. The table was filled using methods of fuzzy logic. A scale (0, 1) for reliability of objects was established, where 0 presents "totally unreliable" parts and 1 stands for "absolutely reliable" objects (Niccolucci et al., 2004). This table is placed in subsection 3.4.1 on page 25 where methods of calculating uncertainty are described in minute detail. However the summary of gained archaeological knowledge about the uncertainty is located in the next paragraph.

Regarding the archaeologists, the remains of the temple and the podium have high reliability value since their parameters are absolutely supported by found evidence. Next, it is clear that the steps could have had two types of performance: the first, and more reliable, includes an ordinary staircase and the second type would have decorative walls along the sides of the staircase. The second option is less reliable because there has been no evidence indicating these side walls. Then, the reliability of walls mainly includes their width and height. The height of the temple depends on the type of the temple whether it is Diastyle (8,0495 m) or Eustyle (8,9965 m), whereas Eustyle has higher uncertainty. Furthermore, the column diameter has high reliability since its dimensions are proved by the fragments of an excavated column's shaft. The diameter of column is then followed by other columns attributes with diverse reliability values. It is known that the columns were Corinthian which means that their capital has the typical appearance with three superposed rows of leaves, but the rest of the column has same parameters as Ionic column. This leads to the fact that the base of an Ionic column can be Ionic or Attic, but an Ionic base is seems to be more suitable for the temple of Diana (Vitruvius Polio, 1914). The next object represents the entablature whose findings covers only one-third of the entire structure, so the final reliability must be lower. In general it is difficult to find evidence for the stone structure, but finding evidence for the wooden structures like doors or timbers is almost impossible, thus they appear as the most uncertain parts. The only characteristic that inferred from the excavation was the width of the doors to the main cella. The roof of the temple can be divided into the main roof and back roof. While the properties of the main roof are more or less certain, there is a lack of knowledge of the back roof. The archaeologists are not sure about its shape thus the two most suitable options are shed or gable roof. Additionally, the shape and dimension of the roof tile is evaluated, followed by the pediment and pediment statues. There are some findings of the structure of the pediment, but almost nothing about the statues which had been probably stolen. Then, the only remnant related to the statue of Diana is its head, thus there is some uncertainty regarding its shape. The next discussed object is the donated statue consisting

of two parts: podium, and statue with its shelter. During the excavation the podium remains were clearly defined unlike the statue for which there is no better information. Lastly the objects of altar, window and antefixes belong to more uncertain parts since archaeologists are only assuming that they might have existed but there are no definite findings.

3.3 *Procedural modelling in archaeology*

It has been proven that procedural modeling is well-suited for archaeological purposes because historical information is often only available in fragments including archaeological remains/ruins, former architectural “rules” derived from other buildings, and additional sources like archaeological texts. Therefore, to achieve the final virtual model, these fragments have to be synthesised (Müller et al., 2006a; Noghani et al., 2012). According to Müller et al. (2006a), this synthesis is predominantly done by human, thus procedural modeling and CGA shapes seems to be a good platform for the following reasons: firstly, procedural modeling offers space for testing hypotheses by adjusting defined parameters. And secondly, the rules are written in a way that they are understandable for humans also and they could be more comprehensive than many pages of textual description (Müller et al., 2006a).

This thesis is based on the CityEngine, software which was successfully introduced into the archaeological field of research (Müller et al., 2006a). The main reasons, why CityEngine was chosen for this project are the possibility of testing several hypotheses, implementation of levels of details and potential solutions for expressing the uncertainty (Haegler et al., 2009). Secondly, this tool has been already verified as a suitable environment by several projects of archaeological reconstruction described in section 2.1. Evolution and characteristics of CityEngine are discussed further in following section.

3.3.1 *Description of the CityEngine*

CityEngine is one of the most mature available procedural modeling software packages. It is a tool allowing efficient modeling of large scale 3D scenes in arbitrary detail (Watson et al., 2008). This tool allows visualization of the relationships of projects, assessment of their feasibility, and examining proposals of the implementation of new projects to reality (ESRI, 2014a). Furthermore, it is flexible in adapting and changing the model characteristics as the knowledge about the modeling site is increasing, for example while visualizing the multiple reconstruction hypotheses of an archaeological excavation site (Haegler et al., 2009).

CityEngine was invented by Parish and Müller (2001) who presented it as the first available system in the domain of procedural modeling of urban environments. Models created in the CityEngine are based on a hierarchical set of rules that can be adjusted according the user’s needs. CityEngine mainly focuses on modeling the traffic network, street furniture and buildings themselves. For road creation, Parish and Müller adapted an already commonly used model of L-systems so it supports generation of large cities. L-system or Lindenmayer system is a type of formal grammar that was primary developed by Astrid Lindenmayer, theoretical biologist and botanist, to describe and model the growth and behaviour of plants (Lindenmayer, 1971). Because of the implementation of enhanced L-system to the CityEngine, it is possible to automatically derive a roadmap from 2D image map inputs and then subdivide the remaining areas to lots and single buildings (Parish et al., 2001). The inputs used may vary from geographical maps (like elevation maps) to socio statistical maps (like

population density or zone maps defining residential and commercial zones). These inputs have the possibility to automatically modify and influence generation of the entire city or single parameters.

CityEngine uses shape grammar called “*CGA shape*” (Computer Generated Architecture) introduced by Müller et al. (2006b). Production rules of this grammar are iteratively evolving a model by adding more and more details. This means that first a volumetric model is created, which is then structured in higher detail as can be seen in figure 19. This is reason why CGA may be called sequential grammar (Müller et al., 2006b). Moreover, CGA is extended by a new type of design grammar called “*split grammars*” that is a powerful tool for generating geometric details on building façades (Wonka et al., 2003).

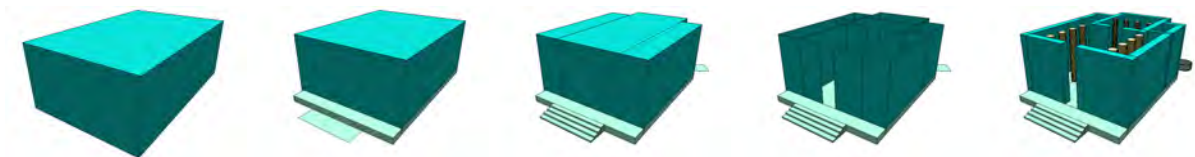


Figure 19: Five consecutive steps of the temple generation using procedural modeling, source: author's work based on Parish et al., 2001

The main focus of this section is not on the street generation and road patterns, but on the modeling of the building's geometry and texture. According to the inventors of the CityEngine, the geometry of buildings is modeled with a parametric and stochastic L-system and every type of building has a different set of rules. The shapes of buildings are determined by their ground plans which could be manipulated by several modules inspired by L-system (Parish et al., 2001). These modules, presenting the main functions, are: transformation (scale and move), extrusion, branching and terminating module, and geometric templates for roof, etc. (Prusinkiewicz et al., 1990). With this approach a high degree of visual complexity can be reached as illustrated in figure 19 which demonstrates the process of building's generation by a series of several consecutive steps (Parish et al., 2001).

A high degree of scene detail can be achieved by implementing procedural textures which capture a certain style of façade by creating a “*style texture*” (Parish et al., 2001). Parish (2001) sorted style textures in hierarchical grids which are based on layered grids. He assumed that façades are composed from several overlayed or nested grid-like structures. “*Layered grids*” is a tool designed for the semi-automatic creation of façades using layering and a simple functional composition techniques. Thanks to this technique complicated textures, including also row irregularities, can be modeled (Parish et al., 2001).

The last thing that should be mentioned while describing CityEngine, are the terminal rules. Haegler et al. (2009) state that terminal rules are presented in cases where the (high detail) shapes are too complicated so they do not result from the composition of other shapes. These rules substitutes the basic shapes (ie cuboid) with premodeled shapes using CAD tools located in a library (Haegler et al., 2009).

3.3.2 Workflow for visual reconstruction of cultural heritage

According to Watson et al. (2008), effective visualization of cultural heritage begins with two steps. Firstly, he stresses that good understanding and knowledge about the architectural elements should be acquired. For the purposes of this master thesis The Ten Books on Architecture written by the

Roman architect Vitruvius Pollio (1914) have been chosen as a main reference for the basic knowledge, because of the textual coding of the building, and sketches of the architectural elements of Roman buildings (Noghani et al., 2012). Secondly, grammar rules have to be created in order to combine desired architectural elements and to create the model. Workflow in the CityEngine environment starts with a certain idea, for instance an archaeological excavation (figure 20).

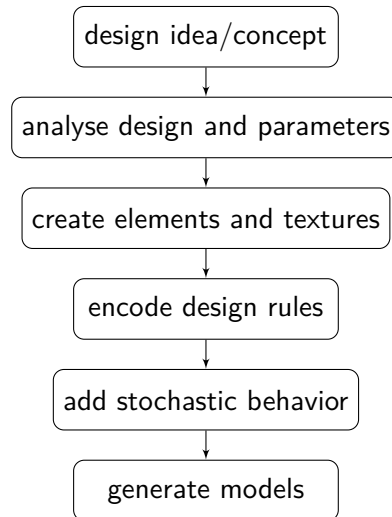


Figure 20: Workflow for a typical architectural procedural modeling project, source: author's work based on Watson et al., 2008

Next, the whole concept has to be analysed so the most important elements and parameters are defined. The best way to create this analysis is in cooperation with the archaeologist. In the case of the Temple of Diana, names of the essential parts are determined from cooperating archaeologists and the literature. Moreover, a table containing all the parameters and values could be established. This analysis should be also focused on the textures and colors. Furthermore, detailed building elements need to be analysed, for instance, decoration features or complex column structures. Then, it should be decided which textures and colors will be used and which building elements have to be modeled separately in other CAD-based 3D modelling software packages. The next step is encoding shape grammar rules in order to create the desired building main structure. After this the stochastic behaviour creating random model parameters can be created if it fits into the concept of the project. This is mainly used for modeling entire cities. As was indicated, the final step is generation of the model. This workflow was also implemented in the project of reconstructing ancient Mayan buildings in Xkipché in Mexico (Müller et al., 2006a).

The shape grammar rules iteratively refine the model in every step by splitting it into smaller parts and adding further details. This process is illustrated in figure 21. This diagram is a simplified example of the generation of a temple model.

3.4 Handling uncertainty and reliability of the 3D model using fuzzy logic

Archaeological features used to be visualized by photo-realistic models only. However the new need for visualizing not only these types of models has increased recently. The essential part of this need is to differentiate between the existing archaeological remains and possible assumptions (Sifniotis et al.,

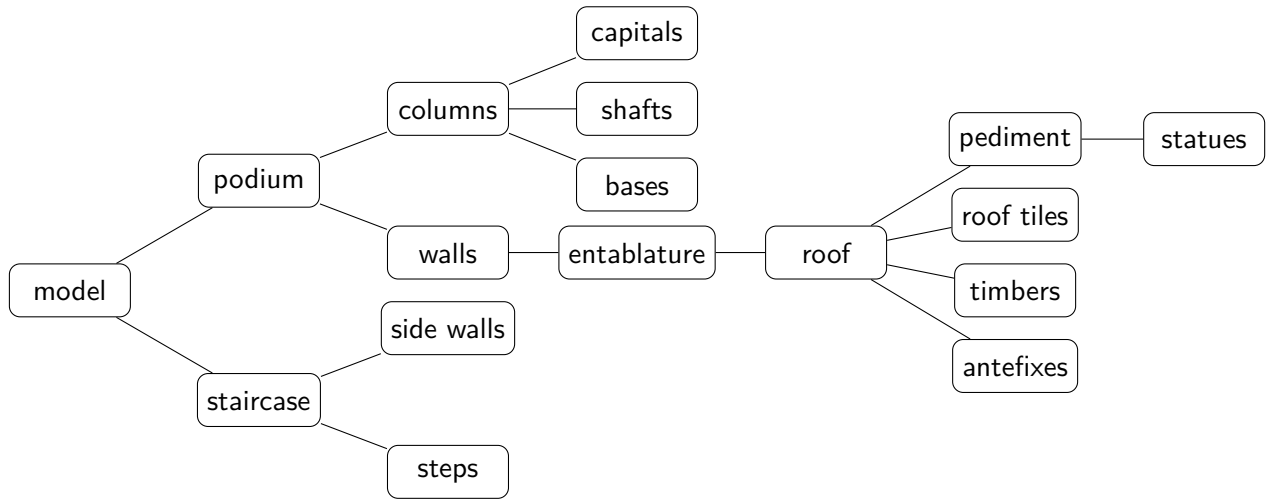


Figure 21: Example of CGA shape workflow of temple

2006). The assumptions correspond to uncertain areas; uncertainty can be defined as “an archaeological expert’s level of confidence in an interpretation deriving from gathered evidence” (Sifniotis et al., 2010, p.1). Implementation of these techniques into the archaeological 3D reconstruction offers new possibilities for using archaeological models, like learning about archaeological hypotheses, or determining and highlighting places where further research is required (Sifniotis et al., 2006).

Handling of uncertainty could be divided in two consecutive parts. The first part of this section deals with the quantifying of uncertainty and the second subsection presents different methods of visualization and communication of the uncertainty to the audience.

3.4.1 Method for calculation of uncertainty

There are several methods using fuzzy logic for calculation of the uncertainty. Each of them has different advantages and disadvantages in terms of complexity or suitable usage. In order to find the best solution for the case study of the Sanctuary of Diana in Nemi, several approaches were combined. Niccolucci et al. (2004) claim that creation of an archaeological model is a stepwise process starting with an initial model M_0 where each step represents addition of a new detail m_n , so at step n a model M_{n+1} is created by adding detail m_n to the model M_n . This description is perfect for combination with the characteristics of procedural modeling used in CityEngine. To define the correct terminology, term “newly added detail” equals to “newly added object”. As Niccolucci suggested, the scale for reliability has been established as an interval $[0, 1]$, where 0 means the most uncertain object that equals a totally unreliable object, and 1 represents the most certain or fully reliable object (Niccolucci et al., 2004). Furthermore, the overall object’s reliability is divided into three components in order to specify the varying uncertainty of its parameters. The idea of the reliability components is based on the correction factors introduced by Tepavčević and Stojaković (2013) and reliability components presented by Niccolucci et al. (2004). The first positional component of reliability $r^{(p)}$ describes the uncertainty of the object’s placement. Secondly, the dimensional component $r^{(d)}$ presents the degree of reliability related to the knowledge of the object’s dimension. Lastly, style reliability $r^{(s)}$ evaluates the style of added objects, for instance the reliability of the structure of the column shaft. Subsequently, the overall reliability of an object r_n is calculated like the minimum of its reliability components as demonstrated on the following formula:

$$r_n = \min(r_n^{(p)}, r_n^{(d)}, r_n^{(s)})$$

Consequently, the calculation of the overall reliability for the entire model $r(M_{n+1})$, defined on the formula bellow, is the minimum of its reliability components and the reliability of the previous model overall reliability r_n . Where r_n is equal $r(M_n)$.

$$r(M_{n+1}) = \min(r_n, r_{n+1}^{(p)}, r_{n+1}^{(d)}, r_{n+1}^{(s)})$$

Numerical evaluation of each reliability component was determined via a formal discussion with the archaeologists, resulting in table 1 containing 21 steps of model generation. This means that 21 objects were stepwise added to the model in order to generate the final virtual reconstruction of the Diana temple. Each of these objects includes the three reliability components described earlier.

ID	Object	Variation	r	$r^{(p)}$	$r^{(d)}$	$r^{(s)}$
0	remains	-	1	1	1	1
1	podium	-	1	1	1	1
2	steps	no sides	0,8	1	0,8	0,8
		with sides	0,5	1	0,8	0,5
3	wall	8,5m (Diastyle)	0,8	1	0,8	0,9
		7,5m (Eustyle)	0,6	1	0,6	0,9
4	column base	Attic style	0,3	0,9	0,9	0,3
		Ionic style	0,8	0,9	0,9	0,8
5	column shaft	-	0,9	0,9	1	1
6	column capital	-	0,7	0,9	1	0,7
7	entablature	-	0,7	0,9	0,9	0,7
8	main roof	-	0,7	0,9	0,9	0,7
9	back roof	shed style	0,4	1	0,9	0,4
		gable style	0,7	1	0,9	0,7
10	roof tiles	-	0,7	1	0,7	0,8
11	pediment	-	0,8	1	0,9	0,8
12	podium donated statue	-	0,9	1	1	0,9
13	main cella door	-	0,5	1	0,8	0,5
14	Diana statue	-	0,4	0,8	0,5	0,4
15	antefixes	-	0,2	0,4	0,2	0,2
16	window	-	0,1	0,2	0,1	0,1
17	side cella door	-	0,1	0,2	0,1	0,1
18	timber	-	0	0,2	0	0
19	pediment statue	-	0	0,3	0,2	0
20	donated statue	-	0	1	0	0
21	altar	-	0	0,2	0,1	0

Table 1: Reliability values of the objects of the Sanctuary of Diana

Additionally, it is necessary to note that in the scope of this thesis only one time frame was used in order to define the reliability. Finally, it is important to mention the significant disadvantage of this method and the solution to minimize this negative effect. The disadvantage lies in the strong dependence of the resulting reliability value on the order in which the details/objects are added. To overcome this issue and enable better results, the objects should not be added randomly. Added objects need to be divided in two groups: construction elements and decoration elements. Then, the construction objects m_{1-11} are added according to the order of the graduating temple reconstruction, for instance walls are added after the podium is generated and before the roof is created. Next, the decoration elements m_{12-21} are added, nevertheless they are ordered according their reliability values so that the more reliable the object is, the sooner it is added. This approach will maintain higher

model reliability in the sequence of the consecutive models for as long as possible.

The reason why this method of fuzzy logic was chosen for the calculation of the uncertainty is following: Despite the disadvantage of this method, which lies in the order of adding new objects to the model, it perfectly suits the environment of procedural modeling. Secondly, the characteristics of fuzzy logic calculation nicely fit to the properties of available archaeological data. Lastly, the formulas of this approach (mainly minimum function) to calculate the reliability value could be easily encoded directly in CityEngine.

3.4.2 *Uncertainty visualization techniques*

Previous section 2.3 gave an overview among the possible approaches of uncertainty visualizations. The main methods were divided into five categories considering using free graphical variables, comparison of several types of models side by side, animation, interactive representation, and integration of external objects. According to Griethe et al. (2006), the decision of which of these methods to choose depends on the purpose of the visualization and it is limited by the technical parameters of given software. That's why the best combination of the above-mentioned techniques should be implemented. Additionally, it should be decided whether the user should be informed only with the existence of uncertainty, or also with the value of the uncertainty (Griethe et al., 2006). These options require different handling and thus dissimilar combinations of visualization methods.

The goal of uncertainty visualization considering the visualization limits of the CityEngine software should be determined in order to choose the suitable visualization techniques. The first goal of the uncertainty visualization of the Diana Temple is to communicate the existence of the uncertainty to the audience. Secondly, it is necessary to express the level of uncertainty of the entire model and its individual parts. Taking into account the possible visualization techniques described in section 2.3 and the purpose of the uncertainty modeling, a unique combination from these methods has been created.

The technical limits of the CityEngine disable using of animation methods. Also integration of external objects is complicated, this has been shown as an disadvantage of the CityEngine since it does not implement annotation which would be helpful for explaining the uncertainty value through 3D text labelling. Next, using textures as a mean of uncertainty visualization would be unduly complex and time consuming while using the CityEngine. It should be mentioned that since the uncertainty of textures has not been implemented in the uncertainty quantification, all visualizations related to uncertainty are without textures, ie medium level of detail. Finally, implementation of edge crispness or line sketchiness seems to be impossible in the CityEngine.

Firstly, the uncertainty is mapped using color visualization schemes. This type of visualization is the most common technique thus implementation of this method should be easily comprehensive for the audience. In order to visually interpret the uncertainty values two schemes has been created; the first Green-Yellow-Red scheme includes changing hue attribute so the greenish tones represents more certain objects and more uncertain objects are colored in reddish tones, as it is illustrated in the following figure 22a. While using this scheme the original model colors need to be overwritten. Secondly, Opaque-Transparent scheme is implemented and alpha parameter expresses the uncertainty of the object. This scheme is outlined in figure 22b, thus the more transparent the more uncertain object. This solution does not require overwriting any colors, that's why the original colors are kept.



Figure 22: Legend for Green-Yellow-Red scheme and Opaque-Transparent scheme expressing the uncertainty

Next, uncertainty is expressed using side by side comparing of different models described in section 2.3. Using this method with a combination of the CityEngine software highlights the advantages of procedural modeling in terms of time-saving generation of multiple models. The third method to map the uncertainty uses the benefits of interactive click-able objects. Fortunately, the CityEngine allows to implement the interactivity through Inspector window or while using the CityEngine Web Viewer. By clicking on desired object, additional information about the uncertainty values appears.

The purpose of this chapter is to cover all the processes necessary to not only generate a 3D model of the Sanctuary of Diana but also to sufficiently fulfil demands and principles related to the visualisation of archaeological virtual reality resulting from the London Charter and the Seville Charter. Consequently generation of the 3D model is an important starting task which is analysed in detail in section 4.1. But it is followed by other processes with same importance. Section 4.2 presents handling of the uncertainty within the meaning of the uncertainty calculation, visualisation and communication. Last but not least, section 4.3 demonstrates achieved results both in the general 3D model generation and in the different renderings and possibilities of uncertainty visualisation. Moreover, it indicates methods of publishing the 3D model and communicating the archaeological knowledge to the public.

4.1 *Generation of the 3D model*

According to the workflow of procedural modeling described in the previous chapter in figure 20 on page 23, the generation of the 3D model starts with analysing the design and determining the parameters of the model. Next, the textures and colors of individual parts of the temple should be defined. Also the elements, called terminate shapes, which are modeled outside of the CityEngine procedural modeling environment should be created. In this case, a free version of 3D modeling software called SketchUp (Trimble, 2014) is used. After description of the generation of the terminate shapes, the selection of colors and textures is presented. Then, the design CGA rules have to be necessarily encoded. Creation of the rules is the most essential part of entire generation of 3D model. The last step included within this section explains implementation of different levels of details (LoDs) and their distinctions. Finally, the 3D model is ready to be generated and published. All these steps are described in more detail in the following subsections.

4.1.1 *Analysis of design and temple's parameters*

The analysis of the temple characteristics has been a challenging issue because many of the parameters are uncertain or unknown even for the archaeologists themselves. That's why this chapter does not only enumerate the parameters but it also explains the way of determining of some of the individual parameter's values. This is also the reason why only the essential parameters are discussed within this subsection, other parameters that are gradually added to the reconstruction rules are only listed in the table located in Appendix A on page 72 containing a complete list of all parameters. The following paragraphs are structured by the larger building elements (podium, columns, etc.), overview among their parameters is given, and calculations of more complicated parameters are explained. The precise values were mainly determined from a SHP and DWG files containing the excavated temple remains. However few dimensions were measured directly at the excavations site during the field trip with TUM to Nemi in September 2014.

Model

First, the values of the temple footprint are defined: the length *modelLength* and width *modelWidth* of the entire area were measured in ArcMap from the above mentioned shapefile (figure 23a). These two values could be considered as the most essential ones because they represent the size parameters of the initial shape used in the reconstruction rules (more information about the rules is in subsection 4.1.4).

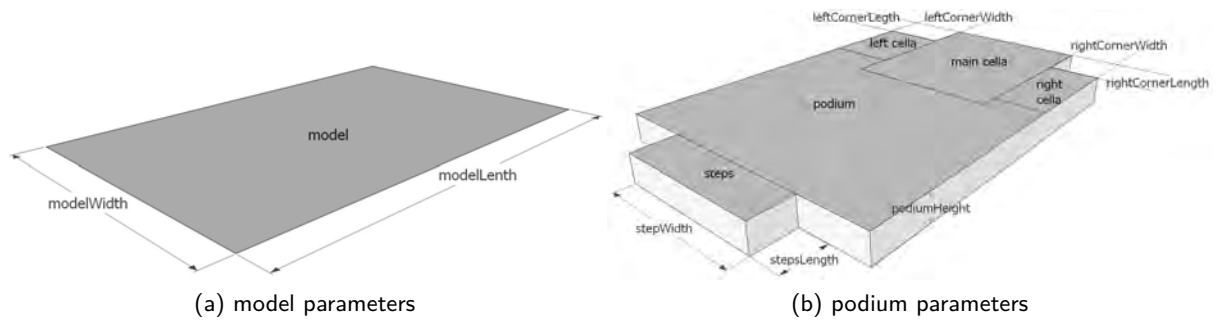


Figure 23: Sketches of the temple's model, and podium dimensions

Podium

In the next step parameters of the podium, which is described as a high platform of the temple, are enumerated. Podium shape is created out of the model by adding more parameters and cutting out the corners formed by cellas, enclosed chambers inside of a temple which may contain the cult statue of the good (Harris, 2005). The cellas are marked as right cella, main cella and left cella so the parameters are clearly defined. The parameters consist of following: *leftCornerWidth*, *leftCornerLength*, *rightCornerWidth*, *rightCornerLength*, *podiumHeight* and *stepsLength*. See figure 23b for the graphical distribution of these parameters. Moreover, the height of the podium was estimated from the available archaeological knowledge and field measurement as described in subsection 4.1.2.

Steps

It is obvious, that the height of the steps *stepsHeight* corresponds to the height of the podium; the length of the steps *stepsLength* can be measured from the shapefile. From the excavations it is not sure whether the staircase had decorative walls along their sides or not, but the archaeologist claims that it was more possible that the steps had no decorative side walls. However if the steps had these walls there are no findings about their parameters. Unfortunately there are also no findings about the parameters of the single step. Thus the parameters are derived according to Vitruvius (1914), who states that the number of steps has to be always odd because if someone puts firstly his right foot on the first step he should also put the right foot as the first on the podium. Furthermore he defines the range for the step height which should be between $\frac{3}{4}$ of Greek Feet and $\frac{5}{6}$ of Greek Feet. Greek Feet is approximately 29,6 cm (Duncan-Jones, 1980) and so the range of heights transferred to meters is (0,222; 0,247). With similar calculation the range for the step depth in meters is derived as (0,444; 0,592). From these ranges it can be determined that there might have been 11 steps and then we can define the height *stepHeight* and depth *stepDepth* of each step by dividing the length and height of the entire staircase by 11 (figure 24a).

Cellas

Another group of parameters is related to the cellas which are characterized as inner closed chambers of a temple which may contain the cult image (Harris, 2005). Namely, the parameters are: *leftCellaLength*, *leftCellaWidth*, *rightCellaLength*, *rightCellaWidth*, and *mainCellaLength* according to the structure described in figure 24b. It should be clarified that all these values were measured in ArcMap from the SHP files.

Walls

The cella parameters are followed by the parameters of the walls. Especially two types of wall widths are defined: *wideWallWidth*, and *narrowWallWidth*, these widths were estimated from the shapefiles

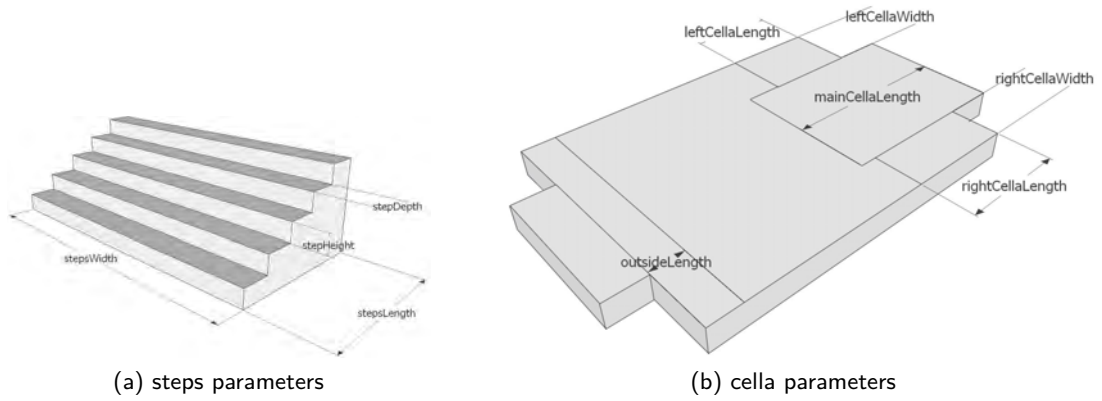


Figure 24: Characteristics of staircase and dimensions of temple's cellas

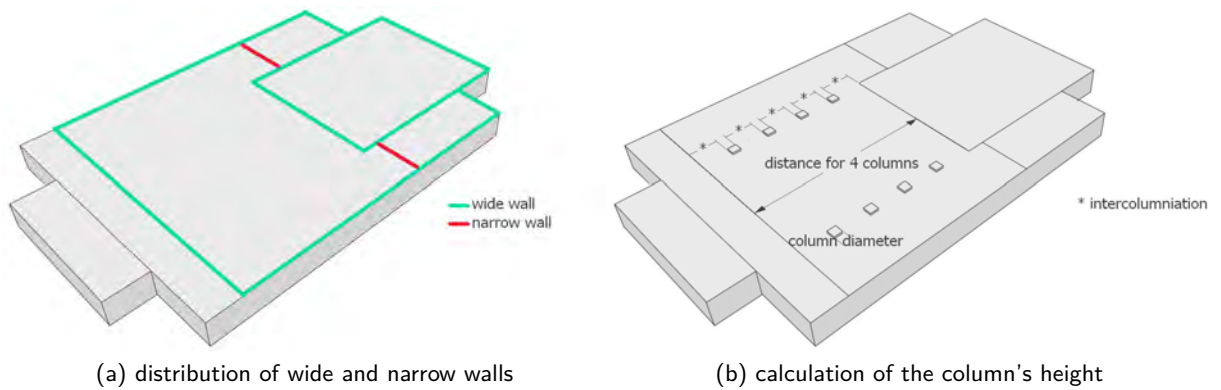


Figure 25: Distribution of walls and explanation of the calculation of the column's height

based on the archaeological knowledge (figure 25a). An important parameter related to walls is the height of the temple called *templeHeight*. This height is a compound of the wall/column height and the height of the entablature. The determination of the precise values of the wall height that equals the column height *columnHeight* is described in the following part related to the columns.

Columns

A column's structure can be divided into three parts: shaft, capital and base. Shaft is the portion of a column between the capital and the base (Harris, 2005). The shapes of individual parts are described in the following subsection 4.1.2 since the entire column structure is composed of terminate shapes. Parameters of individual parts of the column are described in the following paragraphs. Nevertheless, the calculation of the column height is defined first.

Height of the columns

Height of the columns *columnHeight* is, as Vitruvius (1914) states, derived from the column diameter. In general, the height is determined by the style of the intercolumniation. There are five styles of the intercolumniation: Pycnostyle, Systyle, Diastyle, and Eustyle; each of them is presented by different length of the gaps between the columns. In order to find the height of the columns, it was assumed, that according figure 18c on page 19 there were four columns distributed in the certain area. Figure 25b complements this explanation. Since we know the column diameter (0,947 m) and the length of the area where the column's are placed, the calculation is not difficult and the resulting

value of one gap is 2,5808. In other words there might have been 2,725 columns placed within one gap. Taking this result and comparing it to the Vitruvius's definitions of the intercolumniation styles, we get a selection of two styles in which the values are closest to our result. Namely, Diastyle with the gaps wide enough for three columns, or Eustyle in which gaps have a length of 2,25 columns. From this specification there is a last missing step to gain the height value; in Diastyle the height of the columns is 8,5 times the column's diameter, and for the Eustyle it is 9,5 times the column's diameter (Vitruvius Polio, 1914). Thanks to procedural modeling, it is not a complication to have two possible values of the height, hence the users can easily switch between these heights in the Inspector window in the CityEngine. To sum it up, this calculation might be archaeologically imprecise but it gives at least some height value for modeling purposes that is not absolutely out of range.

Shaft

Shaft has two important parameters whose determination should be mentioned. The description is related to the main diameters of the column's shaft *bottomColumnDiameter* and the diminished upper diameter *upperColumn diameter*. Diminution of the shaft's shape towards the upper part of the column was done because of the aesthetic reasons (Vitruvius Polio, 1914). As it was stated before column's diameter is 0,947 m which equals the widest part of the column *bottomColumnDiameter*. However the archaeological findings of the fragments of column's shafts have varying diameters, this supports the phenomena of diminution because there are findings with different diameters. The diameters are ranging between 0,829 m and 0,947 m. Vitruvius (1914) claims that the height of the entire column is needed for the calculation of the column's diminution and determination of the upper column's diameter. As explained above, it is assuming that the height of the column could have 8,0495 m (Diastyle) or 8,9965 m (Eustyle). Now these heights need to be converted to the units of Greek Feet. Resulting in the height between 27 and 30 Greek Feet. Corresponding diminution for this range is reduction by $\frac{1}{7}$ of the bottom column's diameter. Consequently the upper column's diameter is $\frac{6}{7}$ of the bottom column's diameter, hence 0,812 m (Vitruvius Polio, 1914). We can confirm this result with the other findings described in the beginning of this paragraph, while the diameters of all founded pieces of the shafts fit within within the range of bottom and upper diameter (0,812 - 0,947 m).

Capital and base

The remaining parameters of column *columnBaseHeight*, *columnBaseWidth*, *capitalHeight*, *capital-Width*, and last but not least *upperColumnDiameter* are derived from the values of above described bottom and columns diameters by using different ratios according to the notes of Vitruvius (1914). Whereas these elements were modeled outside of CityEngine, their appearance is explained in detail in subsection 4.1.2 containing the description of terminate shapes.

Entablature

Entablature represents horizontal mouldings supported by the columns or walls and it is divided into three elements: the lowest element called architrave, the middle member frieze located between architrave, and the uppermost element named cornice (Harris, 2005). Therefore height of the entablature *entablatureHeight* is a sum of the heights of its inherit parts (*architraveHeight*, *friezeHeight*, and *corniceHeight*) which are also mostly derived from the height of the columns. The height of the cornice is estimated from its preserved fragments. Since the shape of architrave was modeled in SketchUp see the following subsection 4.1.2 for more details.

Roof

Construction of the roof is one of the unclear things about the sanctuary. There were probably two parts of the roof - main gable roof and unknown type of back roof covering the protrusion of the

main cella (figure 26a). The essential parameter of the gable roof is the angle of the roof planes to the horizontal plane (Esri, 2014). According to the literature related to the Roman temples, the angle *mainRoofAngle* was usually 20 degrees (Mark, 1993). Since the back roof shape is unknown, two shapes were implemented. These are gable roof or shed roof whose roof plane is generated with angle *backRoofAngle* to the horizontal plane. The values of the *backRoofAngle* are defined so that the roof aesthetically fits in the model. All the roof planes are divided into the roof tiles with certain width *roofTileWidth* and length *roofTileLength*. Shape of the roof tiles was modeled as a terminate shape and thus the generation of the roof tiles is described in subsection 4.1.2.

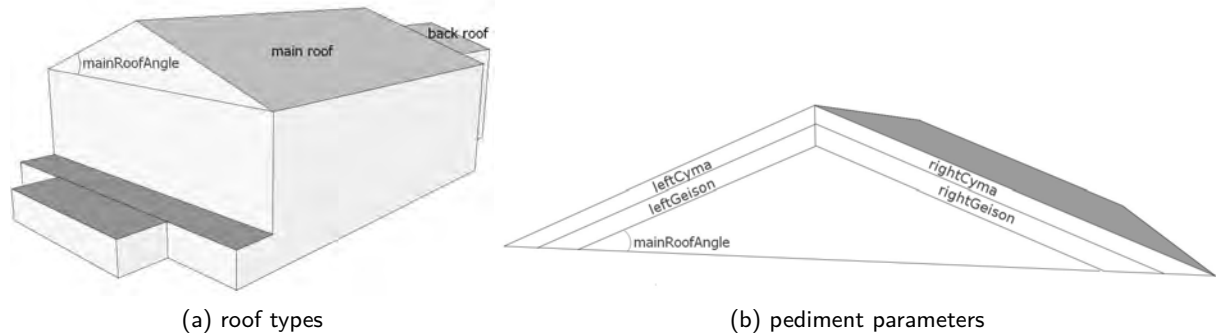


Figure 26: Division of the roof types and demonstration of the pediment structure

Pediment

Pediment forms a triangular gable of the roof which consists of tympanum, geison and cyma (Harris, 2005). Almost every pediment parameter is derived from several ratios described by Vitruvius. The only exception is the depth of the pediment *pedimentWidth* which corresponds to the archaeological findings. Heights of the geison *pedimentGeisonHeight* and cyma *pedimentCymaHeight* are determined from different ratio of the height of the entablature cornice *corniceHeight* (Vitruvius Polio, 1914). Pediment parameters are illustrated in figure 26b and the description of terminate shapes of geisons and cymas are located in subsection 4.1.2.

Doorway

It is clear that this paragraph defines the doorway parameters. Regarding the archaeological knowledge there were two types of doorway. Firstly, the main door leading to the main cella, and secondly there were probably two identical doors to the side cellas. Because the doors were made of wood, it is almost impossible to collect any archaeological evidence. Only knowledge can be derived from the gaps in the walls where the doors used to be. Thus, the width of the main cella *cDoorwayWidth* could have been measured in ArcMap. The remaining values of doorway parameters, like width of the doors of the left and right cella *sDoorwayWidth* and all the doorway heights *cDoorwayHeight*, and *sDoorwayHeight*, are adjusted to the aesthetic perception of the temple. Even though there are two types of doors, the same terminate element is used to represent each of them, see subsection 4.1.2 for more information.

Window

Surprisingly, based on archaeological research there was a window with a glass windowpanes located in the right wall of the sanctuary. Unfortunately this is the only available description regarding the window. Hence the precise placement of a window and parameters of window height *windowHeight* and length *windowLength* are not supported by any evidence. Window element has been also created as terminate shape presented in subsection 4.1.2.

Decorative elements

As it results from the general description of the Roman architecture, the temples always had some decorative elements (Vitruvius Polio, 1914). Unfortunately, there is minimum of findings or knowledge regarding the architectural embellishments of the Sanctuary of Diana. Nevertheless, in order to recreate the temple, some of these beautifying elements should be used. The decoration elements involve: statues decorating the pediment, statue of the goddess Diana situated in the main cella, small features placed along the roof edge called antefixes, altar, and finally the statue including its podium that was donated to the temple and that is placed behind the temple building. Because of the lack of knowledge of the decoration parameters, it is sufficient to only refer to the table in Appendix A on page 72 where all the parameters are listed. Next, it should be referred to the following subsection containing details about individual decoration features.

4.1.2 Modeling of terminate shapes

Second phase of the preparation for the generation of the 3D model is modeling of elements called terminate shapes. As Wonka et al. (2003) state, procedural modeling techniques have limitations in the complexity of architectural details. The suggestion is to create these structures separately and import them to the grammar rules as terminate shapes (Wonka et al., 2006). As it was written in the beginning of this chapter, the terminate shapes are created in SketchUp environment (Trimble, 2014). The generated models are exported to file formats OBJ or DAE which are compatible with the CityEngine environment. As already announced in previous subsection, description of individual elements and their properties is placed in the next paragraphs. The elements are structured more or less in the same order as the temple is generated.

Podium

The first modeled element is the structure of the podium side. Fortunately, there are archaeological findings of a large piece of the podium and that's why the photographs (figure 27a) and field measurements were used to identify the properties of its structure.



Figure 27: Structure of the podium and its components: Francesca Diosono, author's work

It is known that the podium had 4 parts (A, B, C, D) as it is illustrated in figure 27a. The archaeologists assume that the part D had the same structure as the part B but it was turned upside down as it can be seen in figure 27b illustrating the final podium side element including the main dimensions. All other used numerical dimensions are located in the table of parameters located in Appendix A, page 72. Nevertheless, structure of the podium side has unique characteristic since it probably surrounded entire temple apart from the front side. Due to the complex shape of the

temple's cellas, different structures were created to fit in the inner and outer corners of the podium, illustrated in figure 28.

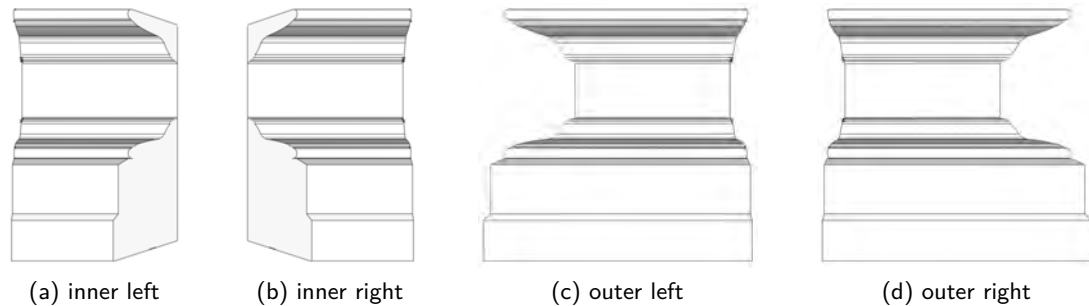


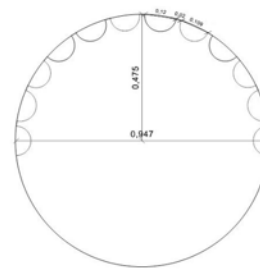
Figure 28: Examples of the podium structures that are adjusted to the inner and outer corners

Columns

As it was stated in the previous subsection, column's structure can be divided into three parts: shaft, capital and base. For the structure of the shaft, there are archaeological findings determining its shape and size, figure 29a serves as an example of one of these findings. Its diameters (bottom and upper) were calculated from the archaeological data using the knowledge of Vitruvius (1914).



(a) example of a finding of the column's shaft



(b) archaeological description of the shaft's structure

Figure 29: Archaeological data for the properties of the column's shaft, source: Francesca Diosono

Next significant characteristic of the shaft is its structure which was created exactly according the archaeological data illustrated in figure 29b. With these dimensions and the values of the diminution the terminate shape of the column's shaft was modeled and it is depicted in figure 30.

The second part of the column, the capital, is the uppermost part which is located above the shaft. *"In classical architecture capitals are one of the most distinctive elements defining the different orders."* (The Doric Order, 2014). The archaeologists found out that the capitals belonged to the Corinthian order, but unfortunately there is no closer evidence of its design. The structure of this type of capital is very complex, and since there is no exact conception of the capital, already created model was downloaded from the SketchUp 3D Warehouse¹. SketchUp 3D Warehouse is an official platform run by Trimble for free uploading and downloading of 3D models in SKP format so they can be further modified directly in SketchUp. The Corinthian capital, with its proportionally placed three rows of acanthus leaves and scrolls, is depicted in figure 31a (Ambrose et al., 2008).

¹<https://3dwarehouse.sketchup.com/>

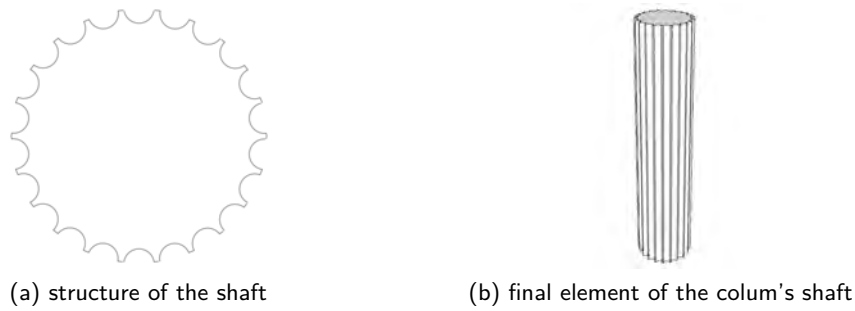


Figure 30: Properties of the column shaft element created in SketchUp

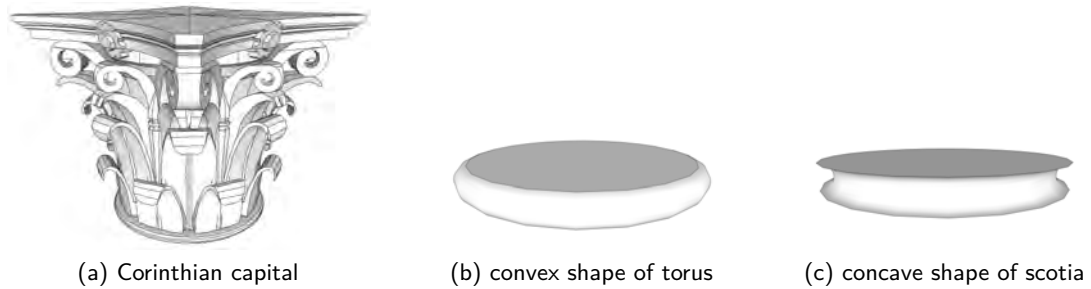


Figure 31: Structures of the column's capital and base components, source: SketchUp 3D Warehouse, author's work

The last and lowest portion of the column is its base. The archaeologist have no convincing evidence about the parameters of the bases so the Vitruvius's (1914) rules were again used. Corinthian columns had the same bases as Ionic columns, and Ionic columns could have had two types of bases: Attic and Ionic. These bases usually consist of various elements structured in different order. These elements could be: plinth, torus, scotia, fillet, and trochillus (Vitruvius Polio, 1914). Plinth is the lowest part of the base which helps to distribute the weight of the column and it has basically shape of a cuboid, hence there is no need of any terminate shape since the build in function in the CityEngine offering insertion of a cube is sufficient (The Doric Order, 2014). The shapes of a torus and scotia are more complex; torus has a semi-circular convex shape (figure 31b) and on the other hand scotia is a concave moulding (figure 31c). Fillet, usually inserted between curved mouldings, has the shape of a cylinder and it was also created beforehand in SketchUp. Finally, trochillus reassembles the concave shape of the scotia and thus the shape of the scotia is also used for the representation of the trochillus (The Doric Order, 2014).

Entablature

This horizontal moulding is divided into the three parts: architrave, frieze and cornice. But it was imported to the CityEngine as one piece of element. However the modeling process is not that simple because of the lack of archaeological findings. There are some pieces of the uppermost part of the entablature among the available archaeological data, thus it is assumed that they are most likely representing the cornice. It was decided to model the elements of architrave and frieze according to Vitruvius's definition and the cornice is modeled according to the excavated evidence.

Cornice is *"a projecting horizontal decorative moulding that crowns a building"* (Ambrose et al., 2008 p.93) and it is a top section of the entablature. Its shape was modeled according the properties

archaeological finding depicted in figures 32a and 32b. Next, figure 32c shows the result of the cornice modeled in SketchUp.

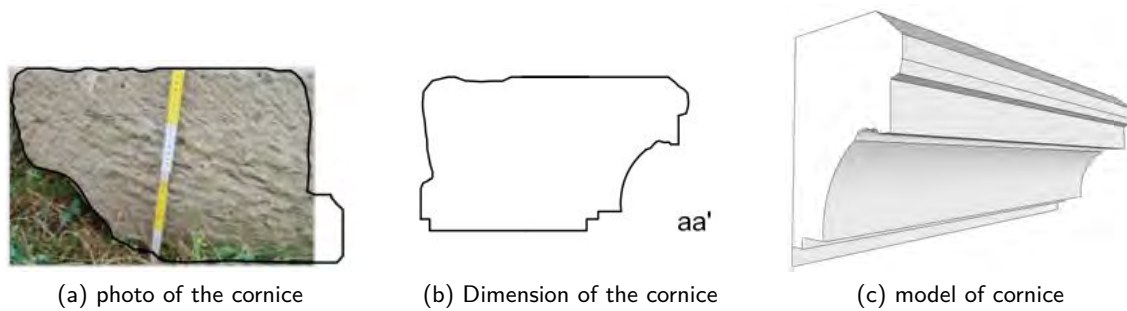


Figure 32: Properties of the cornice, the uppermost part of the entablature, source: Francesca Diosono

Architrave is the lowest member of the entablature and it usually rests on the top of the columns or walls (Ambrose et al., 2008). The third part of the entablature is frieze, which is a decorated central element located between the architrave and cornice (The Doric Order, 2014). Dentils, squared tooth-like blocks are placed over the frieze (Harris, 2005). For both of these parts there is lack of the evidence of their structures, thus their proportions and projections were formed according to the instructions from Vitruvius (1914). Even though some parts of these elements are not that complicated, entire entablature shape was generated outside of the CityEngine. Finished shape of the entire entablature including the cornice part which was modeled according the archaeological knowledge (see previous paragraph) is represented in figure 33a. The images 33b and 33c are demonstrating the generation of the outer corner segments of the entablature.

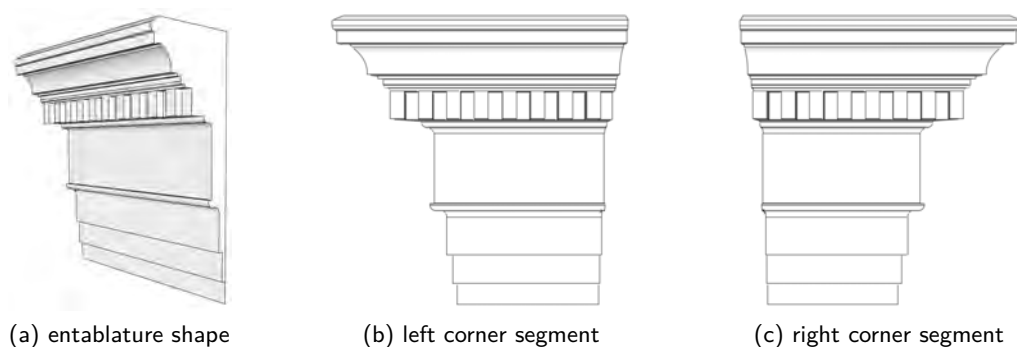


Figure 33: Models of entablature including the corner's structures

Pediment

As it was noted, this triangular gable of the roof consists of tympanum, slanting sides, and cyma. Literature sometimes classifies one more element to the pediment especially a horizontal cornice, but in this reconstruction the horizontal cornice has been already included with the entablature (Harris, 2005). Decoration of the vertical tympanum element is optional, in the case of the Sanctuary of Diana the tympanum was decorated by statues. The only findings of these statues are footprints placed in front of the tympanum on the top of horizontal cornice (figure 34a). Slanting sides called geisons have basically the same structure as the horizontal cornice and they meet in one point at the top (Vitruvius Polio, 1914). Geisons are crowned with moulding named cyma that may serve as eaves and its shape has cuboid proportions (Harris, 2005).

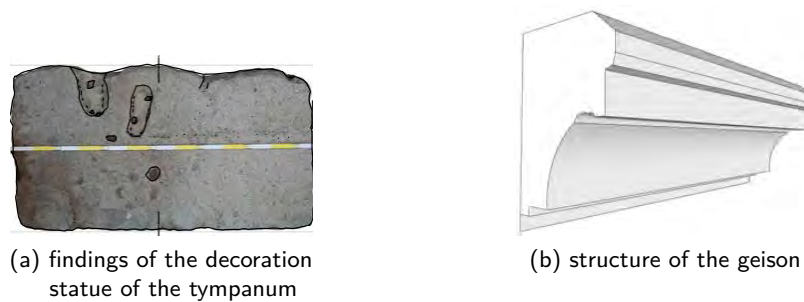


Figure 34: Representation of the pediment elements, source: Francesca Diosono, author's work

In the case of pediment, there is no other archaeological evidence except the footprints of the statues, thus for the structure of geison the exactly same structure was used as for cornice and only simple cuboid shape is used to represent the cyma. The structure of the terminate geison element is visible in figure 34b. Furthermore, as depicted in figure 35, the original shape of geison had to be modified in order to adapt the endings to the triangular shape of pediment, this results in generation of four differently cut shapes.

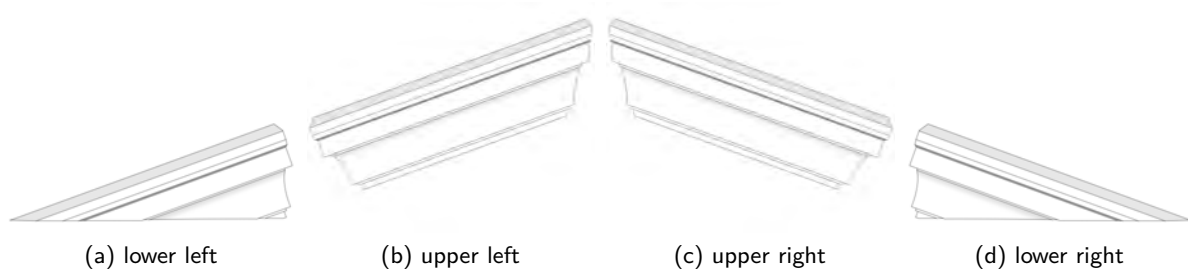


Figure 35: Adaptation of the slanting sides of geison in pediment

Roof tiles

The archaeologists claim that the roof of the Sanctuary of Diana was covered by bronze roof tiles as well as the roof of temple founded in the famous Caligulas ships² recovered from the bed of Lake Nemi in 1929. Sadly, no remains of such tiles were excavated around the temple. Conforming to the archaeological hypotheses the tiles could have been stolen after the temple decayed, since the bronze material can be easily reused. However it was possible to model them accordingly to the pictures of the excavated tiles from the ships (figure 36a). Dimensions of the tiles were approximately estimated and the final tile element is visible in figure 36b.

Doors and window

It is clear that there were door elements in the temple regarding its excavated remains. But it is no surprise that there are no findings of these wooden parts after so many years. The same argument is applied to the window located in the southeaster wall, however there is at least a notion that the window panes were made out of glass. Considering the lack of an evidence, shapes from the SketchUp 3D Warehouse are used for both of these elements and they are illustrated in figure 37.

²<http://www.atlasobscura.com/places/nemi-ships>

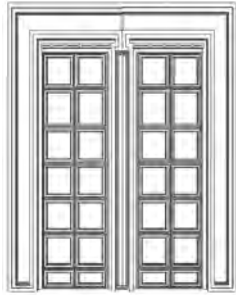


(a) tiles from the excavated ships in Lake Nemi

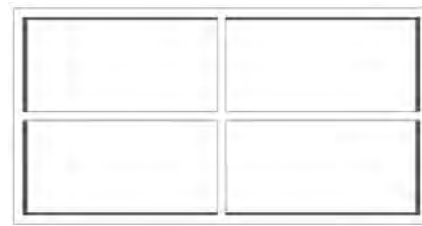


(b) tiles created in SketchUp

Figure 36: Depiction of the bronze roof tiles, source: Francesca Diosono, author's work



(a) shape of door



(b) shape of window

Figure 37: Terminate shapes of door and window, source: SketchUp 3D Warehouse, author's work

Decoration elements

Previous subsection 4.1.1 explained the need of the decoration features and indicated which elements are used to decorate the sanctuary. Yet, here is the enumeration of the mentioned elements: statues decorating the pediment, statue of the goddess Diana situated in the main cella, small features placed along the roof edge called antefixes, altar, and finally the statue including its podium that was donated to the temple and that is placed behind the temple construction.

Statues decorating the pediment

Firstly, three random statues from the SketchUp 3D Warehouse were chosen to decorate the pediment. According to figure 38, it is obvious that they are representing a man, a woman and a bird. Additionally, it is clear that their structures are very simple. There are several reasons for the selection of these statues. First of all, the types of statues were selected arbitrary from the limited choice in the SketchUp 3D Warehouse. Secondly, type of the statues is not that relevant because of the lack of evidence about the decoration - the only known property is a footprint of some statue (figure 34a). Consequently, the simple shapes may turn to an advantage because they fasten the rendering of the entire model.

Statue of Diana

For the second decoration element, the statue of goddess Diana, a solution of using textures instead of terminate shapes is suggested. Since implementation of the randomly shaped statue would miss the essential information that this element is the statue of goddess Diana. Hence statue is presented only with a 2D image of the goddess Diana (see subsection 4.1.3 for more information).



Figure 38: Statues decorating the pediment, source: SketchUp 3D Warehouse, author's work

Antefix

Next, the shape of antefix is presented. Harris (2005) characterize an antefix as a upright slab used as a decoration ornament for the ridge of the roof. Mostly, they are placed at the end of a row of tiles to cover the joints of individual roof tiles (Harris, 2005). Shape of antefix reused from one of the freely accessible tutorials³ provided with the CityEngine software package and figure 39a describes its main structure.



(a) antefix



(b) altar

Figure 39: Terminate shapes of antefix and altar, source: CityEngine, SketchUp 3D Warehouse, author's work

Altar

There has been no findings for element of altar. Archaeologist are assuming that there should have been an altar placed in front of the temple but there are no closer findings about its shape or size. Regarding the notes from Vitruvius (1914), altars should be placed on the lower level than all statues placed in the temple, so those who are praying can look up to the godhead. Furthermore, altar's height is varying according to which good it is dedicated to. In order to simulate the altar, object from SketchUp 3D Warehouse is used. The structure of the altar is depicted in figure 39b.

Donated statue with a podium

Finally, the donated statue with its podium is described. Archaeologists proved that there was a statue behind the temple standing on a circular podium with additional two columns and small roof or pediment. The shape of the podium, and probably also the location of the columns, is distinct from the original archaeological data as indicates figure 17 on page 19. In this figure the podium of the statue is presented by the circle structure in the left upper corner. Thanks to the parameters that can be derived from the data, the podium was particularly modeled as an extra element visible in figure 40a.

³<http://resources.arcgis.com/en/help/cityengine/10.2/index.html>

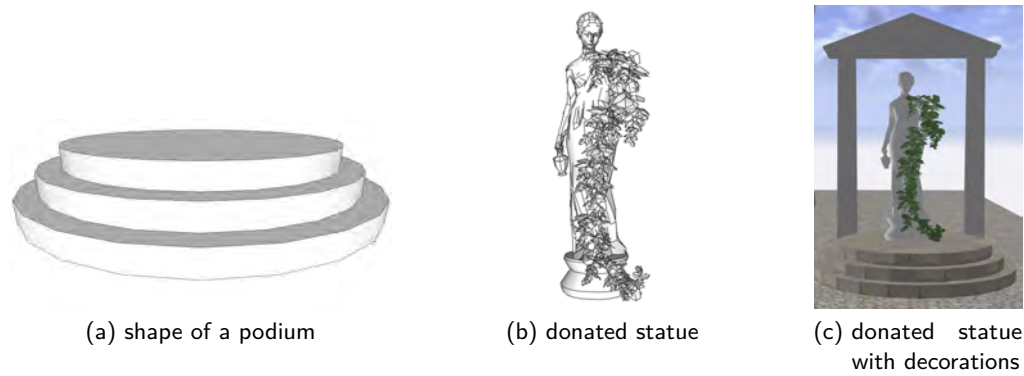


Figure 40: Donated statue placed behind the temple and its elements, source: SketchUp 3D Warehouse, author's work

For the statue itself, there is again lack of information, thus a statue of a woman from SketchUp 3D Warehouse was borrowed (figure 40b). At last, the additional features like columns holding the small roof were formed. As for the statue, there is no closer description neither for this part. That's why simple shapes of cuboid were used and the columns have the same structure as the column belonging to the temple, yet they have different diameter. The overall appearance of the donated statue with its surrounding is shown in figure 40c.

4.1.3 Colors and textures

This subsection presents the selection of colors and textures used for the decoration of the temple. First of all, the used colors are enumerated in the following table 2. They are mostly imitating the materials from which the temple was built, like bronze, stone, white plaster, wood etc. The knowledge about materials was estimated from the archaeological findings or it resulted from the conversation with archaeologist Francesca Diosono. For the description regarding the choice of colors for expressing the uncertainty see the section 4.2.







Color	HEX code	Type of imitation	Usage
	#FFFFFF	white plaster	walls, columns, white mosaic floor
	#EDE6CD	blocks of a stone	podium
	#CD7F32	bronze material	roof tiles, statues
	#854523	wood	timbers, doors, window jambo
	#93B8CC	glass	window glass
	#28272F	black mosaic	black mosaic floor

Table 2: Selection of colors used for the temple reconstruction

Although, the colors used for the reconstruction were just described, there is also need of defining the textures. Textures are used instead of colors in case of high detail resolution, more information about the levels of details is in the subsection 4.1.5. As a source for textures, three web pages were

used (plaintextures.com⁴, LuGher Texture⁵, and TextureKing⁶) in order to find the most suitable illustrations. Images from these pages can be further freely used for non-commercial purposes. Figure 41 is showing few examples of the utilized texture, nevertheless list of all textures and their sources is placed in Appendix B on page 74.

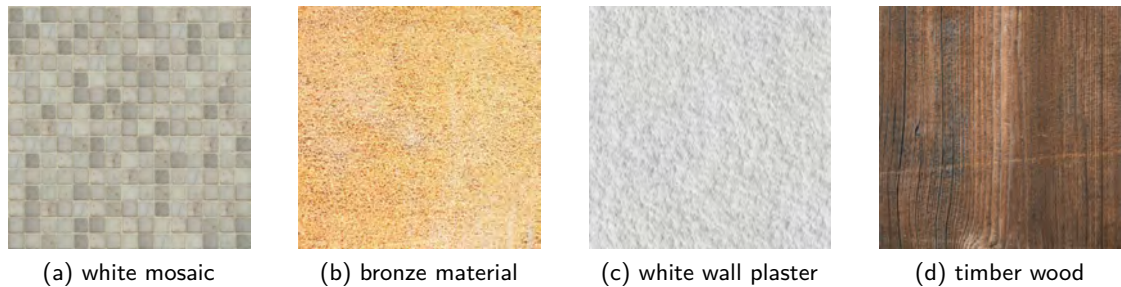


Figure 41: Examples of the textures used for the temple reconstruction, source: plaintextures.com

Finally, as announced in subsection 4.1.2, design of the statue of Diana located in the main cella needs to be discussed. As it was already described, the statue is uniquely formed only by the texture with no terminate shape. The following figure 42a is illustrating the depiction of sitting goddess Diana. The used images of Diana was taken from photobucket⁷ where it is considered as public information thus it can be further reused. Second illustration 42b shows the resulting implementation of the sitting statue in the main cella. The area around the statue texture is transparent thus it appears more natural.

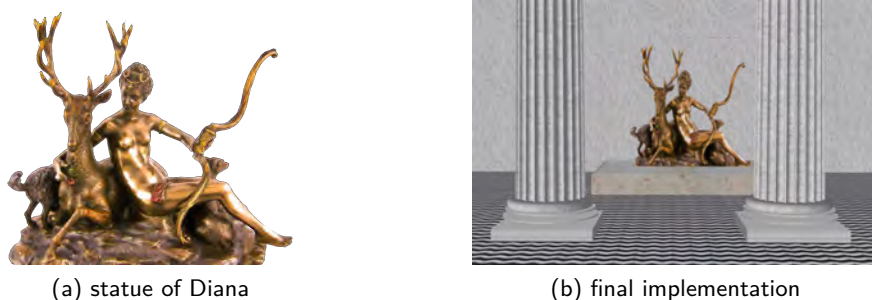


Figure 42: Reconstruction of the statue of goddess Diana, source: photobucket, author's work

4.1.4 Reconstruction rules

This section follows the description of the parameters, terminate shapes and textures. Reconstruction rules presents the most important part of the 3D generation. As it was thoroughly explained in subsection 3.3.1 CityEngine shapes are generated by CGA shape grammar where each generation step is described by a rule and the entire set of rules is then saved within one or more rule files. Thus the following lines are explaining the rules that were created in order to model the Diana temple. However the rules explained here need to be slightly simplified because of their length. Nevertheless, the entire

⁴www.plaintextures.com

⁵www.lughertexture.com

⁶www.textureking.com

⁷photobucket.com

CityEngine project is uploaded on my GitHub account⁸, thus it can be easily explored. Furthermore, I am persuaded that describing every used function would be redundant since the precise definitions are written in CityEngine Help available online⁹. Because of this reason, it was decided to firstly simply describe the main steps of the reconstruction and hence demonstrate the behaviour of procedural modeling techniques. And secondly, highlight those parts of the code, which are interesting, more advanced, or they present the essential feature of archaeological reconstruction.

Explanation of the reconstruction principle

For the generation of the Sanctuary of Diana 131 rules were created. Hence the following lines try to delineate simplified version of the modeling process. For the beginning, two syntax issues need to be mentioned. Firstly in the CityEngine, beginning of the rule is defined by this character `-- >`, but to highlight that following code examples are incomplete and simplified, this character `~~>` is used instead. Secondly, all functions used within one rule are built on the previous ones which means that the first function somehow modify the input shape and the second function's input is the already modified shape, not the starting shape. To avoid this, square brackets `[]` can be used and thus the input for the operations written in the brackets is always same as the starting shape.

The starting rule *Model* is assigned to the footprint of the temple which is further subdivided using function "split" to fractional parts like *Podium* and *StepsArea* as shows the code example below this paragraph. Temple's footprint is afterwards lifted by translate function "t" to form the base for the *Roof*. Next the foundations for the future cellas, named *LeftTemple*, *CenterTemple*, *RightTemple*, are made by "t" and "split" operations. Than the *Pavement* around the temple is created by resizing the initial shape using scale function "s". Finally, starting rule *DonatedStatue* and *Altar* are defined using "t" and "s" functions. All these functions are implementing parameters described in subsection 4.1.1.

```
Model ~~>
[split...Podium]
[split...StepsArea]
[t...Roof]
[t, split...LeftTemple, CenterTemple, RightTemple, OutsideTemple]
[s...Pavement]
[t, s...DonatedStatue]
[t, s...Altar]
```

Regarding the code above, the three dots are indicating that some steps were omitted in order to keep the code simple and clear. Three dimensional shape of a *Podium* is than created by "extrude" function which extrudes 2D shape by certain value. Next among the others, component split function "comp" is applied to decompose the vertical and horizontal faces resulting in horizontal *PodiumFloor*, which is created and further modified, and in the vertical podium sides *PodiumLeft*, *PodiumRight*, *PodiumRightBack*,.... Each of these horizontal faces represents the starting shape for the implementation of the terminate shape of a podium outlined in subsection 4.1.2. An example of an asset implementation is closely characterized in the second part of this subsection named "Remarkable parts of the code".

```
Podium ~~>
split
extrude
comp(f)
PodiumFloor, PodiumLeft, PodiumRight, PodiumRightBack, ...
```

⁸github.com/MariDani/DianaTemple

⁹cehelp.esri.com/help/index.jsp

```
PodiumLeft ~>
  extrude
  i("assets/podium.dae")
```

Since the creation of the steps is more complicated, special paragraph from the next part of this subsection is devoted especially for the steps generation. Areas of the temple's cellas *LeftCella*, *MainCella*, *RightCella* are determined by splitting the main parts of the temple as indicates the following code:

```
LeftTemple ~>
  split
  InnerLeft, LeftCella

CenterTemple ~>
  split
  InnerCenter, MainCella

RightTemple ~>
  split
  InnerRight, RightCella
```

Walls of the temple are modeled by selecting the appropriate vertical faces from the extruded temple parts obtained in a previous step, for example *RightCella*. These faces are further divided into the lower parts *RightCellaWall* and upper parts *EntablatureRightCella*. Lower, but higher, parts represent the walls whose width is gained by extrusion of their faces. The upper parts *EntablatureRightCella* are shorter and they will form the decorative entablature moulding by implementation the corresponding terminate shapes. This is the principle how all the walls in the temple are created:

```
RightCella ~>
  extrude
  comp(f)
  RightCellaWall, EntablatureRightCella

RightCellaWall ~>
  extrude(narrowWallWidth)

EntablatureRightCella ~>
  extrude
  i("assets/entablature.dae")
```

Some of the walls have a doorway included, for instance *MainCellaWall* (that was made equally like *RightCellaWall*). Thus the hole for the doorway *MainCellaDoorway* has to be created again by adopting the "split" operation. Afterwards door's terminate shape is implemented into the slightly modified *MainCellaDoorway* shape. Moreover a new rule called *WallTexture* can be implemented in order to design the wall's appearance. One example of this designing rule is described in the second part of this subsection. The same principle applies for the creation of the window situated in the right side of the temple.

```
MainCellaFrontWall ~>
  extrude
  split
  WallTexture, MainCellaDoorway

MainCellaDoorway ~>
  extrude
  i("assets/door.dae")
```

The starting shape for the colonnade is the same as for the walls. Like *RightCella* was extruded, the *InnerCenter* is extruded in the same way to comp the right and left horizontal faces which are preceding the particular *Colonnades*. Additionally, as for the walls, the entablature part *EntablatureColumn* needs to be subtracted and replaced by the corresponding asset. Subsequently, the *Colonnades* are divided to narrow stripes, from which every second stripe presents the *Column* and the remaining stripes are deleted. Columns are one of the essential features of the Roman temples and that's why an advanced description is located in the next part of this subsection.

```
InnerCenter ~>
  extrude
  comp(f)
  Colonnades, EntablatureColumn

Colonnades ~>
  split
  Column
```

Roof rule created directly from the *Model* is divided into two parts. Firstly, *MainRoof* is a base for the main gable roof and secondly, *BackRoof* forms the back roof covering the protrusion of the main cella. The shape of the *MainRoof* is created by the function "roofGable" modeling a roof with required parameters (ie. roofAngle). Resulting shape is further divided using component split function into the *Pediment* and *RoofTimber*. The same techniques are used for the *BackRoof* implementation of the "roofGable" or "roofShed" operation depending on the user's choice.

```
Roof ~>
  split
  MainRoof, BackRoof

MainRoof ~>
  roofGable
  comp(f)
  Pediment, RoofTimber
```

RoofTimber is a wooden construction holding the future *RoofPlane*. It is created by dividing, resizing and moving different parts of the roof to form the individual beams called *Timber*. These timbers are then replaced with cubes and their design characteristics are added.

```
RoofTimber ~>
  split
  s, t
  Timber
  [RoofPlane]

Timber ~>
  extrude
  i("builtin:cube")
```

Having the timber construction described, the next is the *RoofPlane* that is divided into the *RoofRows* which are scaled, rotated with "r" function, and further split into the individual *RoofTiles*. *RoofTiles* are then adapted for the implementation of their corresponding terminate shape described in subsection 4.1.2. By the adaptation of the tiles, is meant designing their appearance. This code is placed in the second part of this subsection "Remarkable parts of the code" because this is done by more mature functions. *RoofPlane* is also a starting point for creation of *Antefixes* decorating the roof and using premodeled asset as their shape.

```
RoofPlane ~>
```

```

    split
    RoofRow
    [Antefixes]

RoofRow ~>
    s, r
    split
    RoofTiles

RoofTiles ~>
    extrude
    i("assets/roofTile.dae")

```

Last undescribed element of the roof is its vertical part *Pediment* that is extruded by its width and then its faces are separated using component split function. The resulting parts are called *PedimentBottom*, *Tympanum*, *PedimentSideRight*, and *PedimentSideLeft*. Both sides of the pediment, *PedimentSideRight* and *PedimentSideLeft*, are then extruded and prepared for insertion their terminate shapes described in subsection 4.1.2. *PedimentBottom* is cut in pieces which serve as a foundation for the statues decorating the pediment (*StatueMiddle*, *StatueMiddleRight*, *StatueMiddleLeft*,...). Every of these statues is then extruded and substituted with the required terminate shape.

```

Pediment ~>
    extrude
    comp(f)
    PedimentBottom, Tympanum, PedimentSideRight, PedimentSideLeft

PedimentBottom ~>
    split
    StatueMiddle, StatueMiddleRight, StatueMiddleLeft, ...

StatueMiddle ~>
    extrude
    i("assets/man_statue.dae")

```

Next, generation of altar is done using the same functions again. The shape of altar, created in the beginning, is extruded by its height and corresponding terminate element is inserted via the "i" function as describes the code bellow.

```

Altar ~>
    extrude
    i("assets/altar.dae")

```

Finally, the last decoration elements has to be defined. Explanation of the statue of the goddess Diana is placed within the next part of this section. Thus shape for *DonatedStatue*, created in the beginning, is described as a last feature of the temple's reconstruction. The generation of all its parts: *BackStatuePodium*, *BackStatue*, *BackStatueShelter*. *BackStatuePodium*, and *BackStatue* is created by extrusion and implementation the corresponding asset files. *BackStatueShelter* forms certain shelter or roof connected to the statue and its final shape was created by several further rules such as *ShelterColumnBase*, *ShelterColumn*, *ShelterRoof*, or *ShelterPediment*.

```

DonatedStatue ~>
    [BackStatuePodium]
    [BackStatue]
    [split...BackStatueShelter]

BackStatuePodium ~>
    extrude
    i("assets/statue_podium.dae")

```

```
BackStatueShelter -->
  ShelterColumnBase, ShelterColumn, ShelterPediment,...
```

Remarkable parts of the code

Following paragraphs are devoted to the explanation of several specific sections of the reconstruction rule. The described parts were selected because their sophistication or execution differs from the rest of the code. Contrary to the first part of this section named "Explanation of the reconstruction principle", this part represents non simplified rules. In the beginning, as announced earlier, steps creation will be described. First of all, conditional rule is used to determine whether the steps have or do not have the side walls. In both cases the resulting rule is written as *Steps (no_of_steps)*. This entry differs from the all rules mentioned so far because of the brackets behind the name of the rule. This encoding is called parametrized rule and it passes an attribute defined in the brackets from the first rule to the second one (Esri, 2014). Parametrized rules are applied a lot within the reconstruction rules, since they are very effective and furthermore they also limit redundant parts of the code. So the rules of shapes which are modeled by exactly same function only with different parameters, do not have to be repeated. In the case of steps generation, parameter defining number of steps *no_of_steps* is passed. Subsequently, the essential rule for steps creation *Steps* refers to the number of steps only as to *n*. From the case-else statement, it results that generation of steps lasts as long as *n* is bigger than 0. When this condition is fulfilled a single step is extruded, next its shape is reduced by the depth of the steps *stepDepth* and the rule *Steps* is called again, but with the difference that number of steps *n* is scaled down by 1. This guarantee that the steps generation will finished after forming desired number of steps.

```
StepsArea -->
  case Type_of_Steps == "no sides" :
    Steps (no_of_steps)
  else :
    split (z) {stepSideWidth : StepSide
              | ~1 : Steps (no_of_steps)
              | stepSideWidth : StepSide}

Steps (n) -->
  case n > 0 :
    extrude (world.y, stepHeight)
    split (x) { stepDepth : comp(f) {all : StepsTexture}
              | ~1 : Steps (n-1)}
  else : NIL
```

Explanation of next rule covers the implementation of terminate shapes described in subsection 4.1.2. The principle of inserting of such an element is simple: a mass model is created, for instance a cube that presents a bounding box for the asset which is placed and scaled within this cube (Esri, 2014). The following code presents implementation of the podium structure depicted in figure 28 on page 34. As it was described there are special assets for the corners areas of the podium, illustration of integration of this type of assets is in the code below. Firstly, the podium part, in this case *PodiumBack*, is divided by "split" function to the corner parts and the middle part. Corner areas are named according their location, thus *PodiumOR* refers to outer right corner and *PodiumOL* presents the outer left corner. *PodiumC* stands for the center part of the podium. Secondly, each of these sections is modified to fit well to the temple and corresponding terminate shapes is inserted using "i" operation.

```
PodiumBack -->
  setPivot(xyz, 1)
```



```

t(-podiumOffset,0,0)
split(z) { 1 : s('1, '1, scope.sz+podiumAssetOffset) PodiumOR (0)
          | ~1 : PodiumC(0)
          | 1 : s('1, '1, scope.sz+podiumAssetOffset)PodiumOL (0)}

PodiumOR (setPivot) -->
  extrude(podiumOffset)
  setPivot(yzx, setPivot)
  i("assets/podiumOR.dae")
  comp(f) {all: PodiumTexture}

```

Columns are the fundamental features in Roman architecture, thus the reconstruction of the column is examined in the following lines. Firstly, the entire *Column* is replaced by a cube that is further split to its fractional parts specified in subsections 4.1.1 and 4.1.2. As it was stated, the sizes of the column fractions are derived from its diameter, hence the column's diameter is again passed to the subsequent rules to define the corresponding dimensions. Secondly, the individual fractions are modified and replaced with the terminate shape as the rules *Capital* and *Shaft* are demonstrating. Creation of the column's *Base* is more complicated since there are two types of the bases and moreover the base is further divided to its components, described in the subsection 4.1.2, which are afterwards also replaced by the adequate 3D elements. To achieve this the rule needs to be longer and it would be redundant to place it here since it is developed by same principles and using the basic operations.

```

Column(diameter)-->
  t(0, 0, -diameter/2)
  i("bultin:cube")
  split (y) { columnBaseHeight : Base(diameter)
             | ~1 : Shaft
             | capitalHeight : Capital(diameter)}

Capital(diameter) -->
  s (diameter*6/7, '1, diameter*6/7)
  s (1.45, '1, '1.45)
  center (xz)
  i("assets/corinthian_capital_withAbacus.obj")
  ColumnTexture

Shaft -->
  i("assets/shaftColumn.dae")
  ColumnTexture

```

As it was announced in previous part of this subsection, generation of roof tiles should be explained. Remarkable characteristic of its reconstruction is resizing and rotating the *RoofRows* in order to establish an overlay and skew of the roof tiles. Then the roof rows are divided into the individual *RoofTiles*. Secondly, roof tiles are substituted with their premodeled elements introduced in subsection 4.1.2. Unlike the previously described rules, implementation of the bronze texture *bronze_tex_rand* is demonstrated in the case of roof tiles. Texture needs to have exactly set up projection parameters. However unique characteristics of the tile's texture is the establishment of the material shininess. Shininess is included in order to imitate the bronze material and value 20 out of Phong specular exponent (0, 128) is used (Esri, 2014).

```

RoofRow(scale_y) -->
  s('1, 'scale_y, '1)
  r(scopeCenter,-1.8,0,0)
  split (x) {~roofTileLength : RoofTile}*

RoofTile -->
  setPivot(zxy, 0)

```



```

i("assets/roofTile.dae")
s('1, '1, '1.045 )
texture(bronze_tex_rand)
setupProjection(0, scope.xz, roofTileWidth, roofTileLength)
projectUV(0)
set(material.shininess, 20)

```

Additionally, definition of the *bronze_tex_rand* parameter should be explained. Contrary to the other texture parameters used in the temple reconstruction, more images of bronze textures are used in order to create realistically looking heterogeneous imitation of the bronze roof. The first line of the following code ensure random selection of a tile texture out of these images. The second line illustrates the difference between the random selection and common texture definition using single image.

```

bronze_tex_rand = fileRandom("assets/Textures/bronze*.jpg")
const bronze_tex = "assets/Textures/bronze4.jpg"

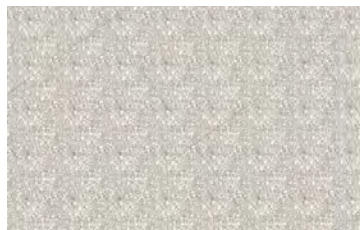
```

In general, default texture projection to large areas sometimes results in unwanted uniform patterns that are visible in figure 43a. Following code shows the prevention of this unwanted phenomenon on the example of the pavement creation. Each squared texture tile is randomly rotated by 0, 90, 180 or 270 degrees. To enable the rotation of the textures, the entire area needs to be divided in individual rectangles that equal to one tile. The resulting texture is shown in figure 43b.

```

textureAngle = 25% : 0 25% : 90 25% : 180 else :270
Pavement -->
  setupProjection(0, scope.xz, 3, 3)
  texture(irrStone_tex)
  projectUV(0)
  rotateUV(0, textureAngle)

```



(a) no rotation



(b) random rotation

Figure 43: Pavement reconstruction with and without rotation of the texture tiles

Lastly, the generation of the main cella doorway is included within this explanatory part. Firstly, the designated wall, in this case *MainCellaFrontWall*, needs to be among other operations split, in order to create the *Doorway* area with its appropriate width *cDoorwayWidth* and height *cDoorwayHeight*. Than, the *Doorway* rule is modified in order to be ready for the insertion of doorway element described earlier in subsection 4.1.2 using "i" function. Finally, all the faces of the asset are segregated by component split operation in order to further doorway designing.

```

MainCellaFrontWall -->
  extrude (wideWallWidth)
  split(z) {~1 : WallTexture
    | cDoorwayWidth : split (y) { mDoorwayHeight: Doorway

```

```

| ~1 : WallTexture}}
| ~1 : WallTexture }
Doorway -->
  setPivot(xyz, 1)
  set(trim.vertical, false)
  i("assets/door.obj")
  comp(f) {all : DoorwayTexture}

```

4.1.5 Implementation of levels of detail (LoD)

Thanks to the characteristics of procedural modeling, it is possible to implement different levels of detail (LoDs) within one rule file. LoDs are encoded as a different parameters of an attribute and hence CityEngine user has the opportunity to switch fast between the individual levels in the Inspector window and thus adapt the visualization to his/her needs. The temple reconstruction provide three different LoDs enumerated in the following lines, which are based on the distribution of the LoDs for CityEngine (ESRI, 2014b):

- *LoD 1 - Low Detail*

This LoD shows only the colored mass model without any textures as it is displayed in figure 44a. The implementation of terminate shapes is mostly omitted and for the essential parts of the temple, like columns, the original detailed assets are replaced with simple cylindrical or cuboid elements. This view could be used for the navigation of camera or the basic perception of the temple's characteristics. Since this model is loaded very fast it does not require any special hardware.



(a) LoD 1 - Low Detail



(b) LoD 2 - Medium Detail

Figure 44: Demonstration of low and medium LoD

- *LoD 2 - Medium Detail*

Contrary to the low level of detail, medium LoD does not omit the terminate shapes and that's why the user is getting the complete idea of the sanctuary's visual appearance. LoD 2 includes all the statues, structure of columns, podium, etc. The colors remains the same as in LoD 1. As it can be seen on the figure 44b, this representation is considered as a standard for the scientific analysis and exploration of the temple.

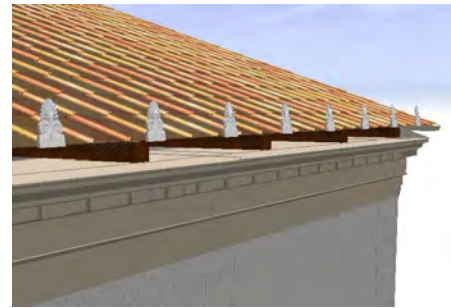
- *LoD 3 - High Detail*

The last and the most impressive visualization is provided by LoD 3. Contrary to LoD 2 it is enriched by various textures of materials. The patterns of the textures are more visible while zooming in. Thus this visualization should look as the most realistic one, almost photo realistic (figure 45). However photorealism might become a disadvantage as it is described in

section 1.1.3. Despite this criticism, LoD 3 has been involved because of three reasons. Firstly, it complements the previous mentioned LoDs, next it shows the extensive possibilities of the CityEngine, and additionally it shows the temple from the different point of view which may be an advantage. Since this model includes lot of assets and texture it requires more powerful hardware for developing the temple than usual laptop.



(a) LoD 3 - High Detail



(b) roof bronze tiles and white wall plaster

Figure 45: Presentation of the high LoD

4.2 Modeling uncertainty of the 3D reconstruction

Uncertainty of the Diana temple is modeled using fuzzy logic approach described earlier in section 3.4. This section is structured into two parts: firstly, subsection 4.2.1 describes the calculation of the uncertainty of the model and it provides explanatory table and figure. The second subsection 4.2.2 presents the methods of uncertainty visualization which are demonstrated on several figures. Both of these subsections are enriched with explanations of how was the uncertainty calculations and visualizations implemented directly in the code of the reconstruction rules. This implementation enables management and visual feedback of the object's reliability values directly in the environment of the CityEngine. The essential parts of the code ensuring calculation and visualization of uncertainty are described in the following subsections. Finally, it is important to mention that the textures do not take part in the uncertainty calculation, thus the assessment of the model reliability corresponds to the LoD 2; version of the model including all terminate shapes but omitting the material textures.

4.2.1 Calculation of uncertainty

The calculations were done accordingly to the description of the fuzzy logic method in subsection 3.4.1. Reliability values of the temple's objects were obtained from the formal conversation with archaeologist as described in subsection 3.2.3. These reliability values indicate that there is a sequence of 21 models ($M_0 - M_{22}$) representing different stages of the temple generation. The results of the uncertainty calculation is demonstrated in the following table 3.

The first two columns of this table are indicating the sequence number of the appropriate model and the name of added detail. The third column defines possible variation of the added objects. Corresponding reliability values r_n to the individual objects and their variation is placed in the column named 'r'. Different object's variations occur in four cases thus the final temple model may take 16 different forms whose reliability values $r(M_n)$ are represented in the remaining columns. The table implies that

ID	Name	Variations	r	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
0	remains	-	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	podium	-	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	steps	no sides	0,8	0,8	0,8	0,8	0,8	0,8	0,8	0,8	0,8	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5
3	wall	with sides	0,5																
		8,0495 m	0,8	0,8	0,8	0,8	0,8	0,6	0,6	0,6	0,6	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5
		8,9965 m	0,6																
4	columnBase	Attic style	0,3	0,3	0,3	0,8	0,8	0,3	0,3	0,6	0,6	0,3	0,3	0,5	0,5	0,3	0,3	0,5	0,5
		Ionic style	0,8																
5	columnShaft	-	0,9	0,3	0,3	0,8	0,8	0,3	0,3	0,6	0,6	0,3	0,3	0,5	0,5	0,3	0,3	0,5	0,5
6	columnCapital	-	0,7	0,3	0,3	0,7	0,7	0,3	0,3	0,6	0,6	0,3	0,3	0,5	0,5	0,3	0,3	0,5	0,5
7	entablature	-	0,7	0,3	0,3	0,7	0,7	0,3	0,3	0,6	0,6	0,3	0,3	0,5	0,5	0,3	0,3	0,5	0,5
8	mainRoof	-	0,7	0,3	0,3	0,7	0,7	0,3	0,3	0,6	0,6	0,3	0,3	0,5	0,5	0,3	0,3	0,5	0,5
9	backRoof	shed style	0,4	0,3	0,3	0,4	0,7	0,3	0,3	0,4	0,6	0,3	0,3	0,4	0,5	0,3	0,3	0,4	0,5
		gable style	0,7																
10	roofTiles	-	0,7	0,3	0,3	0,4	0,7	0,3	0,3	0,4	0,6	0,3	0,3	0,4	0,5	0,3	0,3	0,4	0,5
11	pediment	-	0,8	0,3	0,3	0,4	0,7	0,3	0,3	0,4	0,6	0,3	0,3	0,4	0,5	0,3	0,3	0,4	0,5
12	podiumDS	-	0,95	0,3	0,3	0,4	0,7	0,3	0,3	0,4	0,6	0,3	0,3	0,4	0,5	0,3	0,3	0,4	0,5
13	cellaDoor	-	0,5	0,3	0,3	0,4	0,5	0,3	0,3	0,4	0,5	0,3	0,3	0,4	0,5	0,3	0,3	0,4	0,5
14	DianaStatue	-	0,4	0,3	0,3	0,4	0,4	0,3	0,3	0,4	0,4	0,3	0,3	0,4	0,4	0,3	0,3	0,4	0,4
15	antefixes	-	0,2	0,2	0,2	0,2	0,2	0,2	0,2	0,2	0,2	0,2	0,2	0,2	0,2	0,2	0,2	0,2	0,2
16	window	-	0,1	0,1	0,1	0,1	0,1	0,1	0,1	0,1	0,1	0,1	0,1	0,1	0,1	0,1	0,1	0,1	0,1
17	sideCellaDoor	-	0,1	0,1	0,1	0,1	0,1	0,1	0,1	0,1	0,1	0,1	0,1	0,1	0,1	0,1	0,1	0,1	0,1
18	timber	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
19	pedimentStatue	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
20	donatedStatue	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
21	altar	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 3: Calculation of uncertainty for the stepwise process of the temple generation

the model's reliability values $r(M_n)$ are in general decreasing with increasing amount of added details. However decrease of the reliability values depends on the selected variation of the objects. Nevertheless, the reliability values of the 6 most detailed models $r(M_{16-21})$ are balanced because of the high uncertainty of added details. An illustration of the all possibly resulting models with their reliability values $r(M_n)$ is depicted in following figure 46. The figure enhances the resulting reliability value by coloring the appropriate rectangles where green represent the most certain models with the highest reliability values and contrary red colors are assigned to the models with the lowest reliability values.

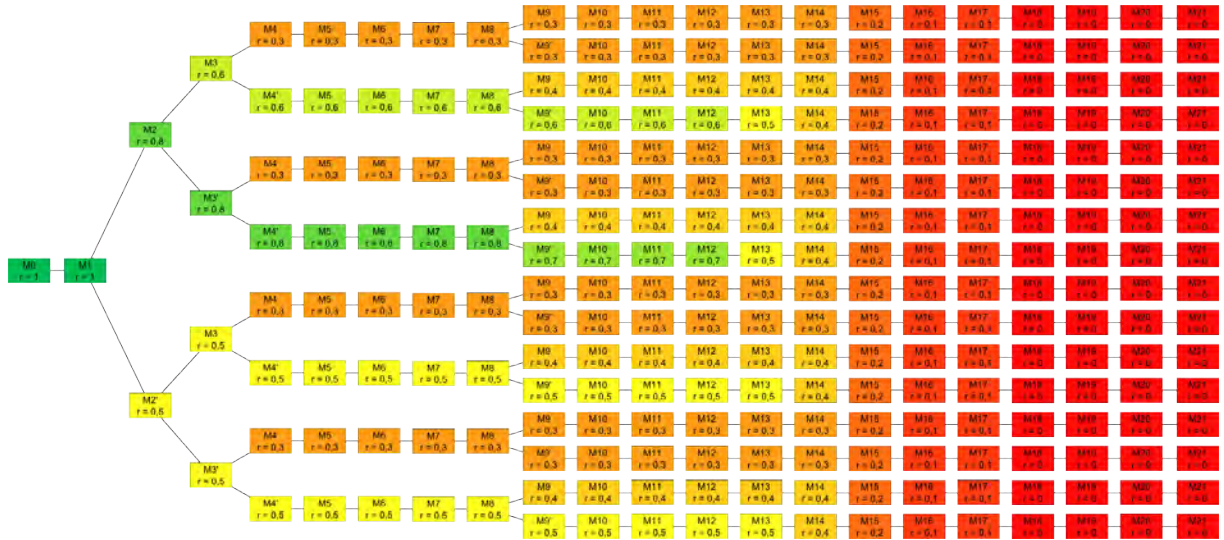


Figure 46: Variations of final temple models with their reliability values

The assessment of the sample containing 16 resulting models would be useless with using the most detailed models. In order to enable meaningful assessment of such a sample, adding of decoration

elements $r(M_{12-21})$ needs to be omitted. The division of the objects to the construction and decoration elements is described in subsection 3.4.1. This will form a comprehensive set of models where a comparison of the reliability values $r(M_{11})$ can show relevant results (see table 3). The visual representation of this set of models is illustrated and described in the following subsection 4.2.2. While comparing these models the most reliable model is determined as the model number 4 with no sides of along the staircase, height of 8,0495 m (Diastyle type), columns with Ionic bases and gable shaped back roof (see table 3).

The implementation of the uncertainty calculations into the CityEngine is divided into three parts. Firstly, the reliability values are declared as attributes even with their variations as shows the following example:

```
attr M0_remains = 1
attr M1_podium = 1
attr M2_steps =
    case Type_of_Steps == "no sides" : 0.8
    else /*"with sides"*/ : 0.5
attr M3_wall =
    case Temple_Type == "Diastyle" : 0.8
    else /*"Eustyle"*/ : 0.6
```

The advantage of declaring variables as attributes is that user can easily modify their values in the Inspector window in the City Engine as it was explained earlier in subsection 4.3. Interconnectivity of the individual parts of the reconstruction rules ensures that any user modification will immediately change the reliability value for the entire model and thus its visual appearance. The second and equally important part is definition of the minimum function which was encoded in cooperation with Matt Buehler, CityEngine machinist.

```
findMin(valList, index, min) =
    case (index < 0): min
    else:
        case (float(listItem(valList, index)) < min): findMin(valList,
            index-1, float(listItem(valList, index)))
        else: findMin(valList, index-1, min)

findMin(valList) = findMin(valList, listSize(valList)-1, float(listItem(
    valList, listSize(valList)-1)))
```

The last and necessary part is calling the minimum function with appropriate values. All models with sequentially added objects have differently modified code in order to calculate the correct reliability value. For instance, to calculate the reliability value $r(M_3)$ the code looks like the following example.

```
reliability = (findMin(str(M0_remains) + ";" +
    str(M1_podium) + ";" +
    str(M2_steps) + ";" +
    str(M3_wall) + ";"))
```

4.2.2 Visualization of uncertainty

Visualizations of the uncertainty should communicate the existence of the uncertainty to the audience and express the level of uncertainty of the entire model and its individual parts. This subsection describes the final results of the implementation of the visualization methods described in subsection 3.4.2.

Firstly, two created color schemes are used: Green-Yellow-Red scheme and Opaque-Transparent scheme. Greenish colors represents the more certain features and on the other hand reddish colors are used for uncertain objects. Regarding the transparency scheme, the transparency increase with increasing degree of uncertainty. The following figure 47a shows the implementation of the Green-Yellow-Red scheme. The next figure 47b displays the final rendering of Opaque-Transparent. It is visible that the more transparent the more uncertain object.

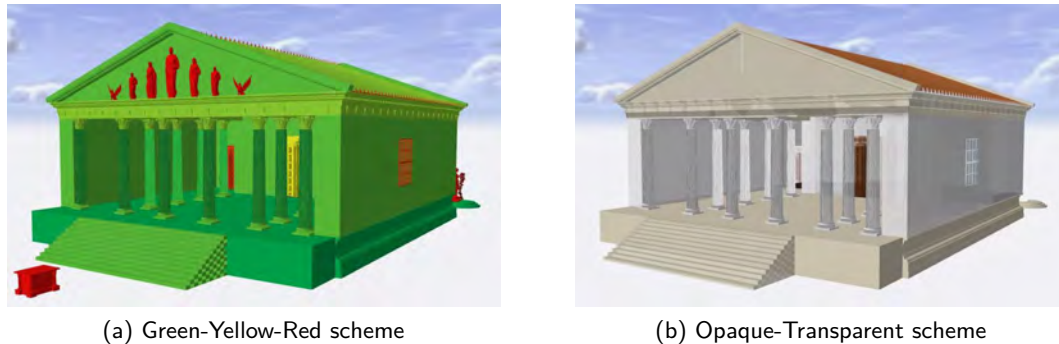


Figure 47: Uncertainty of parts of the temple expressed by different visualization schemes

It is clear that these two schemes are not mutually exclusive, thus they could be combined and used together. However such a combination would not bring any additional scientific value. Furthermore it would be redundant while using CityEngine solution for publishing the models on-line which has benefits for the uncertainty representation. This advantage lies in the environment of the CityEngine Web Viewer that offers among the others a swipe tool that swipes between different layers/visualizations. Thus the user can easily compare for instance Green-Yellow-Red color scheme model with Opaque-Transparent scheme showing uncertainty of model's individual parts (figure 48). Other benefit of the swipe tool relates to the usage of transparency visualization which may be sometimes questionable since the absolutely uncertain objects appear as "ghosts". However this characteristic may turn into advantage while using the swipe tool because than it is clear which objects are missing, thus which objects are least certain to happen (see the pediment statues in figure 48). The other tools and possibilities of the CityEngine Web Viewer are described in subsection 4.3.1.

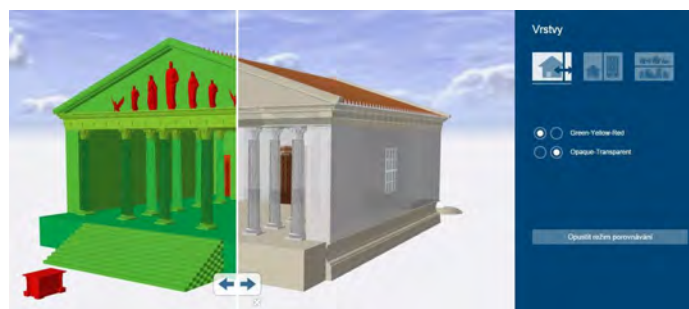


Figure 48: Swipe tool in CityEngine Web Viewer comparing uncertainty of parts of the temple illustrated by different visualization schemes

Second way of expressing the uncertainty is side by side comparison of different models. This visualization consists of 16 different models. This number has been concluded from the available archaeological information while different possibilities of four parameters were combined. These

parameters are: type of steps (with and without side walls), height of temple that corresponds to the type of temple (Eustyle, Disastyle), type of the column's bases (Attic, Ionic), and shape of the back roof (shed, gable). This method is suitable for models whose varying parameters are easily visually distinguishable, unfortunately it is almost impossible to distinguish the varying parameters in the case of the temple of Diana. Thus even this visualization method is enhanced with Green-Yellow-Red scheme and Opaque-Transparent scheme as illustrated in figure 49. This enhancement helps user to understand that there are several possibilities of the model while each of these option has different uncertainty value.

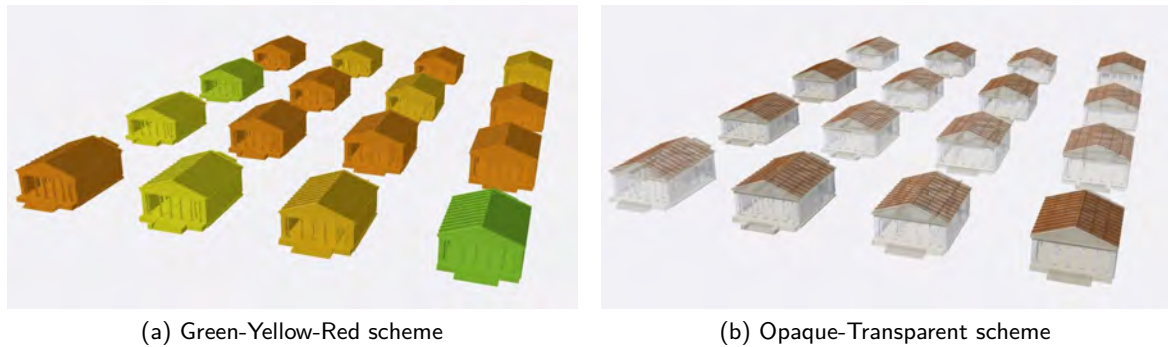


Figure 49: Visualization of different variations of the Diana temple using color schemes to enhance the value of reliability

The third technique used to communicate more information about uncertainty is via the click-able objects. By clicking on desired object, additional information about the precise uncertainty for the entire model and for its individual parts appears. Figure 50 illustrates rendering additional information in the CityEngine in the Inspector, the main characteristics and usage of the CityEngine Web Viewer environment is discussed in more details in section 4.3.

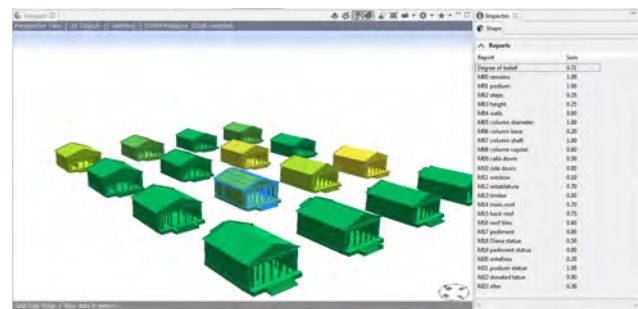


Figure 50: Using interactivity in order to enhance uncertainty visualization in CityEngine

As it was stated earlier in section 3.4.2 CityEngine does not support any labels, moreover it does not offer any solution for the explanation of the used colors schemes ie legends. These shortages are caused by the fact that it is still a relatively recently published software whose functions and possibilities are increasing with every new release. Thus we can only hope that these issues will be solved in near future. Meanwhile the only possible way how to display a legend is to project picture including the legend on a newly created shape "on the ground" and create a bookmark for the view which is zoomed on this shape (see figure 59b on page 60). It is obvious that this is not the best way but does not offer any more sophisticated solution so far.

Last but not least, the principle of implementation of these visualizations into the reconstruction rules is explained in general. First of all, the color schemes definition should be described. The Opaque-Transparent scheme does not require any special definition since it is originally implemented in the CityEngine reconstruction rules. However since a special Green-Yellow-Red color scheme has been created, it needs to be declared as follows:

```
uncertainty_color(reliability) =
  case reliability >= 0      : "#ff0d0c"
  case reliability >= 0.1    : "#ff3d11"
  case reliability >= 0.2    : "#ff6e15"
  case reliability >= 0.3    : "#ff9e19"
  case reliability >= 0.4    : "#ffce1c"
  case reliability >= 0.5    : "#ffff21"
  case reliability >= 0.6    : "#cdf12b"
  case reliability >= 0.7    : "#9be536"
  case reliability >= 0.8    : "#66d741"
  case reliability >= 0.9    : "#33ca49"
  else /*reliability == 1*/ : "#00bc55"
```

Then each of the model's objects is provided with its own special color and transparency values which are varying dependently on the type of visualization. For better understanding see the following code example of the podium reconstruction. The first case differentiates individual parts of the temple according their reliability values as depicted in the figure 47 by color scheme *U_Part_Colors* or transparency scheme *U_Part_Transparency*. In this case the corresponding reliability values are assigned to each object's color *Podium_Color* or transparency value *Podium_Transparency* directly without any calculation.

```
Podium_Color =
  case U_Colors      : uncertainty_color(reliability)
  case U_Part_Colors : uncertainty_color(M1_podium)
  else               : "#EDE6CD"

Podium_Transparency
  case U_Part_Transparency : M1_podium
  case U_Transparency      : reliability
  else                     : 1

Podium -->
  set(material.opacity, Podium_Transparency)
  color(Podium_Color)
```

The second case includes visualizations which are showing the final model reliability value $r(M_n)$ thus the color hue or the degree of transparency is same for the entire model as illustrates figure 49. In this case the reliability value related to the entire model and calculated with the previously explained minimum function is assigned to the object's color *Podium_Color* and transparency value *Podium_Transparency*. Finally, the following code example demonstrates the usage of the report function that enables the view of the additional information after clicking on desired model as shows the previously mentioned figure 50.

```
Model -->
  report("Degree of belief", reliability)
  report("M00 remains", M0_remains)
  report("M01 podium", M1_podium)
```


4.3 Results of the reconstruction

The results of the reconstruction of the Sanctuary of Diana are divided into four separate scenes while each of them brings different kind of knowledge. These scenes can be opened in the CityEngine and viewed by the archaeologists offering them a possibility to change certain attributes and "play with the model" without deeper knowledge about procedural modeling, or encoding of the reconstruction rules. The characteristics of these scenes are structured in the following paragraphs.

Realistic temple reconstruction

The first scene offers a realistic 3D model of the temple including the three different levels of details. Moreover there are layers with the 3D models of generated excavation remains which can be visually classified by the type of the remains. Next, the scene covers satellite picture of Nemi surroundings and detail photograph of the excavation of the Diana temple. Lastly, there is a layer with the plan of the entire excavations in Nemi area showing other excavated buildings and features. Settings of the layers in this scenes offers the possibility of changing parameters of certain attributes directly in the Inspector window in order to test new hypotheses or to explore different dimensions of the temple's objects without the necessity of modifying the code. These attributes are for instance: changing between the levels of detail, turning on and off the appearance of the roof and timber in order to see the inside of the temple, turning on and off the side walls along staircase, or opening and closing doors (see figure 51).

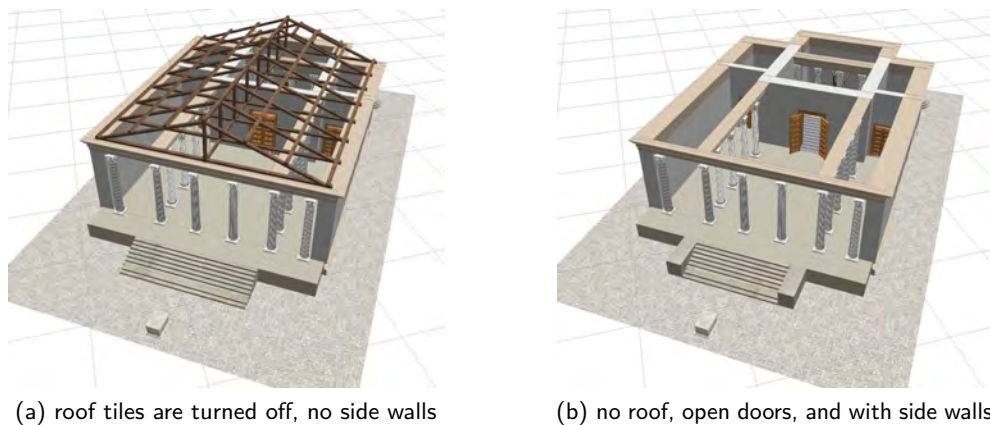


Figure 51: Modifying different parts of the temple using Inspector window in CityEngine

Next, user can easily change attributes like height of the temple, diameter of columns, size of the roof tiles, changing the number of the steps in staircase, etc. by only inserting desired value in the Inspector window. Figures 52a and 52b show different modification of the roof angle and the height of the temple. Lastly, figure 52c presents the simple control of the Inspector window.

Uncertainty of the temple parts

This scene presents the visual reconstruction of the uncertainty of individual temple's objects. As it was described in section 4.2.2 there are two schemes showing the degree of reliability: Green-Yellow-Red and Opaque-Transparent (figure 47 on page 53). The interaction of this scene lies in the possibility of changing the object's reliability values directly in the Inspector and hence gaining differently rendered model. This function might be used to demonstrate new findings about the uncertainty of temple's

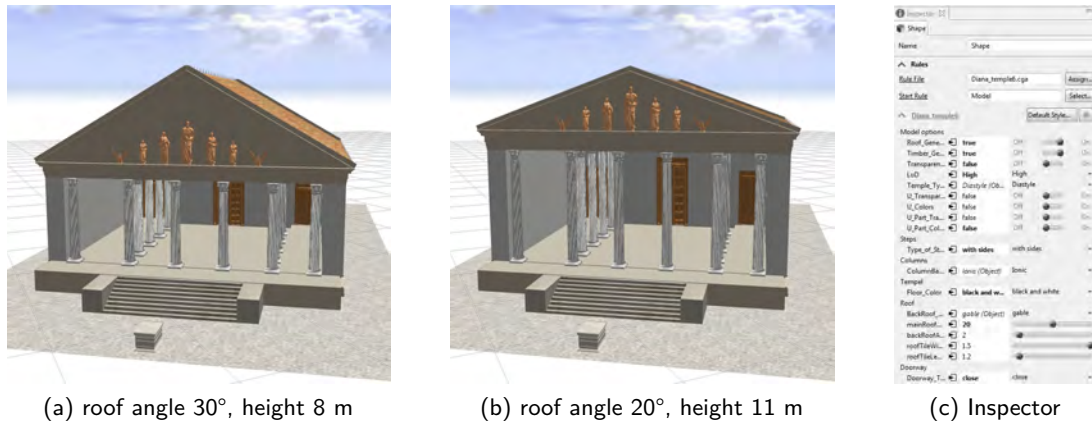


Figure 52: Modifying parameters of the roof angle in the Inspector window in CityEngine

objects. The following set of figures 53a and 53b shows how simple is the modification of the values and automatically obtaining new visualization.

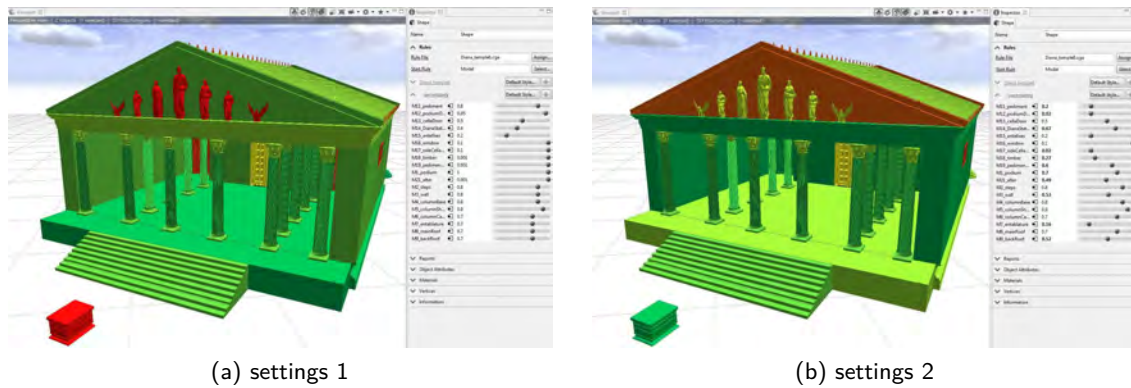


Figure 53: Changing of the reliability values via the Inspector window in CityEngine

Set of 16 temples

The scene containing set of 16 different temples was also already mentioned in section 4.2.2. It exploits the benefits of the visualizing method "side by side" described in subsection 3.4.2. Contrary to the previous scene the color rendering is not for the individual objects, but it shows the final reliability value for the entire model. All 16 temples have different objects parameters and thus they have varying reliability value. As well as the previous scene this scene can be controlled in the Inspector, specifically the reliability values of individual objects can be updated and thus the final model's reliability changes and so does its visual rendering. An example of this modification is depicted in figure 54

Sequence of temple generation

This scene presents the generation of the Diana temple with parameters corresponding to the highest possible value of reliability. There are all stages M_{0-21} of the gradual temple modeling, starting with the excavated remains which are absolutely reliable and continuing with models with gradually added details and progressively decreasing reliability. Thus the user gets familiar with the process of temple

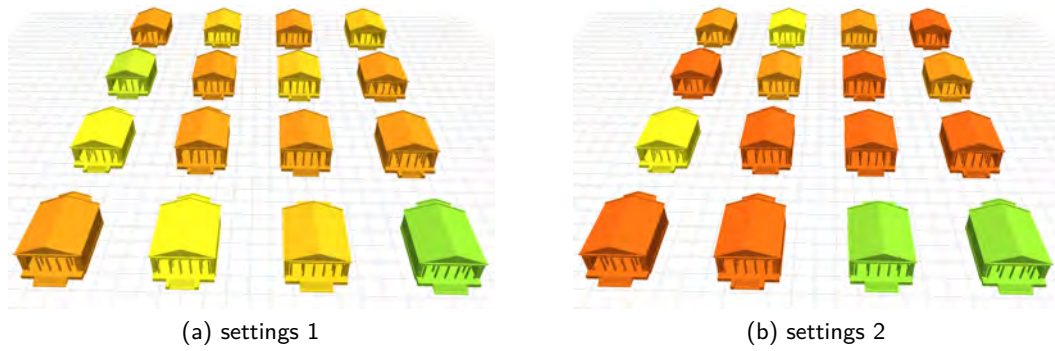


Figure 54: Changing of the reliability values via the Inspector window in CityEngine

generation and simultaneously with the effect of increasing uncertainty with increasing amount of added details. The sequence of temples results in a row of differently rendered temples as illustrated in figure 55.

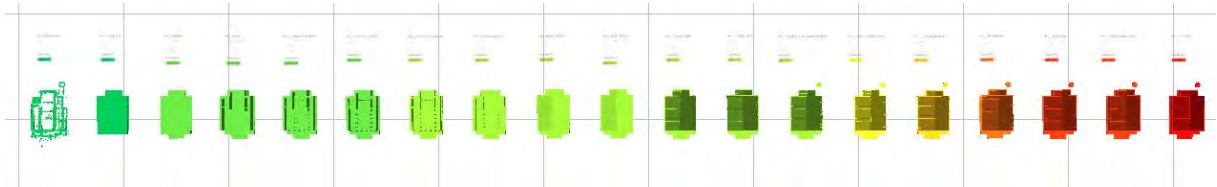


Figure 55: Sequence of the temple generation, beginning with certain remains on the left and ending with detailed and less certain model of the temple on the right

Additionally, to ensure that the user will have with enough information there are labels placed next to every temple containing more information. Labels are containing name of the model M_n , all reliability components of the added details (positional $r_n^{(p)}$, dimensional $r_n^{(d)}$ and style $r_n^{(s)}$ reliability) and finally the overall reliability value for the model $r(M)_n$. Unfortunately, there is no option to create labels or legend for thematic layers in the CityEngine thus the only option was to create a new layer including labels that are placed on the ground behind the temples. Moreover this layer also contains legend explaining the color scheme used for expressing the uncertainty.

4.3.1 Publishing the model online

Finally, to make the scenes described above accessible to the public, they have been published by the CityEngine Web Viewer. CityEngine Web Viewer is a web application based on HTML5 and WebGL technology and thus there is no need for any plugins to visualize and publish the 3D content (Esri, 2012). The main function of the Web Viewer is visual exploration of the 3D scenes which is enhanced by several tools. The first set of tools (zooming, panning and rotating) helps user to navigate across the scene. Then, there is a possibility of turning on and off different layers. Next, info window panel shows more information about objects after clicking on them. The most powerful tool is the comparison environment which enables to compare two different layers. It has three modes while each of them is using swipe tool between selected layers. Additionally, the Web Viewer has a search function that enables searching through the objects and attributes. Lastly, user can mod-

ify the general scene setting: light, shadows, setting an exact date, or he/she can export a print screen.

To publish the scenes, they needed to be exported to the CityEngine Web Scene format (3ws) which is a web-optimized format that can be viewed with the CityEngine Web Viewer (Esri, 2014). The export was done in the CityEngine environment and the scenes were uploaded via the ArcGIS online account. Moreover special web page has been created to fulfil the demands of the London Charter and the Seville Charter about providing enough documentation. This page¹⁰ presents the 3D reconstruction of the Temple of Diana. Firstly, it shows the overview of the created scenes so users can navigate to them and open them in the CityEngine Web Viewer (figure 56a).

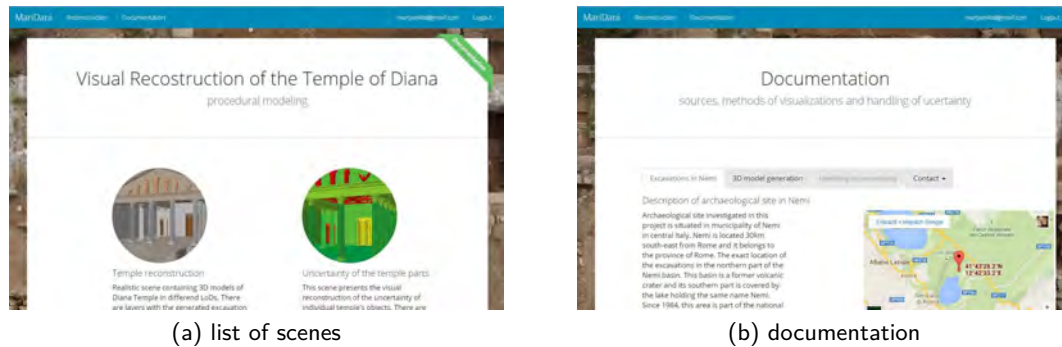


Figure 56: Web page containing all required information about the reconstruction

But especially it contains the required documentation about the sources, process of 3D creation, and explanation of the reliability of the generated model (figure 56b). By including the documentation section users should be provided with enough information for good understanding of the entire reconstruction process and its results. Access to the web page is protected by password because some of the archaeological data used for the temple reconstruction can not be published yet. The published scenes and benefits of using Web Viewer are described in following paragraphs and illustrated on the following figures.

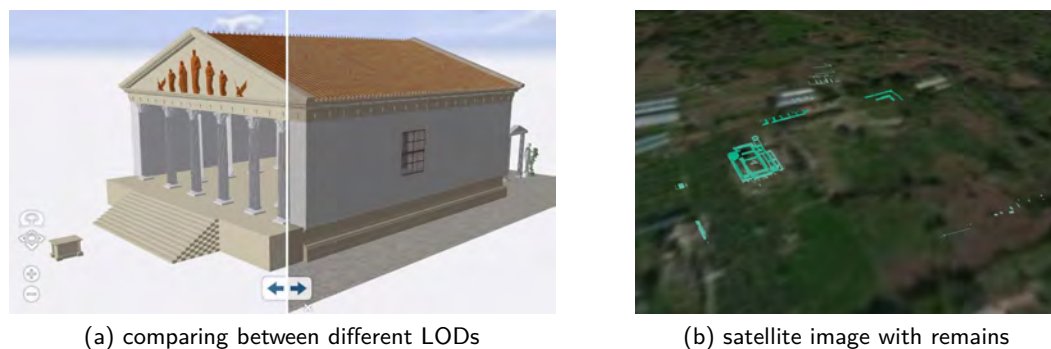


Figure 57: CityEngine Web Viewer, scene containing realistic temple reconstruction

The first scene containing realistic temple reconstruction is presented in the following figures. Figure 57a illustrates the built in swipe tool while it compares between two different levels of detail: the

¹⁰<http://dianatemple.herokuapp.com>

medium LoD2 and high LoD3. The figure 57b shows the excavation site with the satellite image and 3D reconstruction of the excavated remains. The types of excavated remains are differentiated using special color scheme based on the results form previous field projects in Nemi. Legend for this color scheme is also included in this scene. Although the appearance of the second scene presenting the reliability of individual temple parts has been already shown, the following set of figures 58 serves as an example of using different camera position via bookmarks. Bookmarks are one of the benefits of the Web Viewer; they are reusing bookmarks that were created in they CityEngine. These bookmarks are then exported with the scene content and hence user can easily access them in the web scene, furthermore a camera fly can be done using these bookmarks.

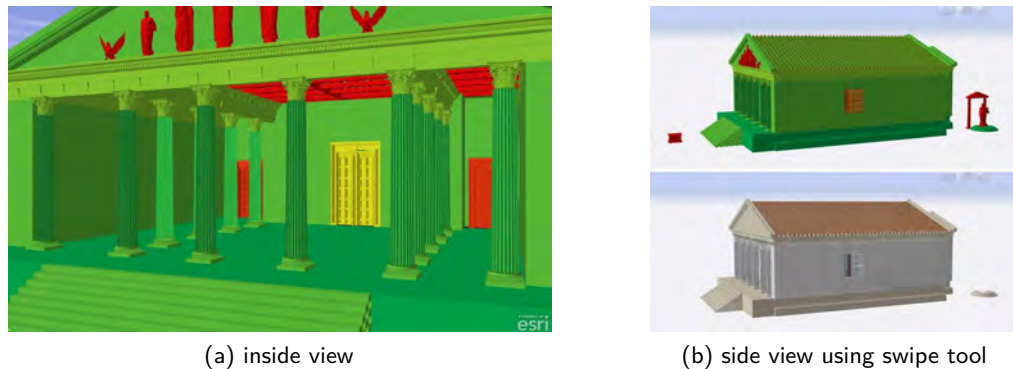


Figure 58: CityEngine Web Viewer, scene showing the uncertainty of the temple parts expressed by different visualization schemes

Thirdly, scene covering the set of 16 temples enables the user to switch between realistic representation and uncertainty representation. Furthermore, figure 59a serves as an example of the information window which appears after clicking on selected temple. The information in the window allows the user to compare the reliability and other characteristics of individual temples. The reliability comparison is then enhanced with the color scheme visualizations. Figure 59b demonstrates the placement of the legend of the Green-Yellow-Red color scheme.



Figure 59: CityEngine Web Viewer, example of the set of 16 temples and corresponding color legend

The last published scene displays the sequence of the temple generation. This is is mostly done using the above mentioned bookmarks that allows movement from one step of the generation to another. The models are available in three different renderings: realistic colors, uncertainty expressed by colors,

or uncertainty expressed by transparency. These renderings allow the user to take the advantage of the swipe tool and compare between different visualizations. The scene contains a layer of labels which belong to the generated models. These labels show the information about the overall reliability of the model and compositional reliabilities of currently added object/detail. Following figures 60a-f depict several examples of the different stages of the generation process.

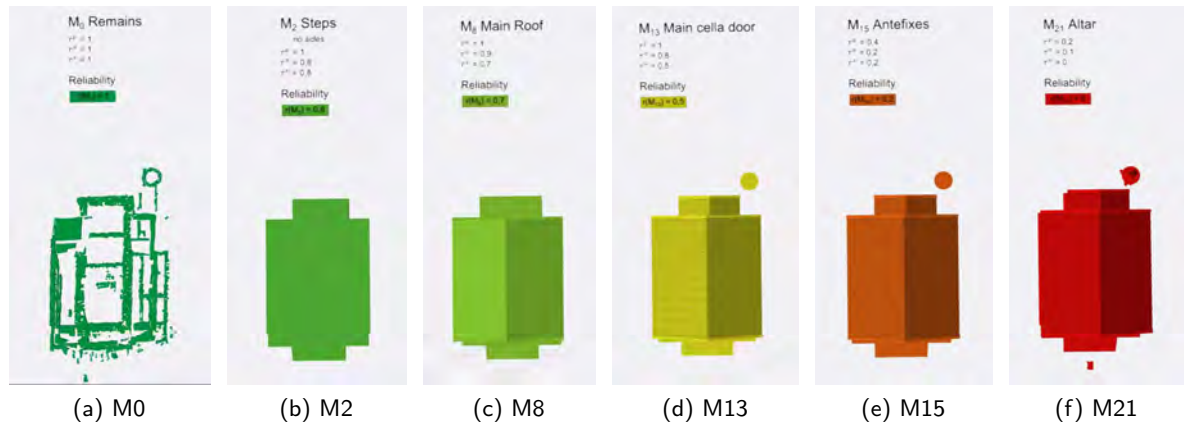


Figure 60: CityEngine Web Viewer, few examples from the sequence of temple's model generation

This chapter gives space for the discussion of the results achieved in this thesis that were previously described in subsection 4.3. Next, the methods used for the 3D model generation are discussed and their main advantages and disadvantages are outlined. Characteristics of these results are also compared with the outcomes of the recent studies defined in chapter 2. The first aim of this thesis delineated in section 1.4 is the reconstruction of the archaeological data of the Sanctuary of Diana at Nemi using procedural modeling techniques in the CityEngine software package. The second goal is calculating uncertainty of the generated model by using fuzzy logic theory. The value of the uncertainty should be then visualized in the logical and understandable way. The main challenges that were determined in the beginning of this thesis are also discussed in the following paragraphs.

Procedural modeling and CityEngine

It has been proven that procedural modeling is suitable for modeling archaeological virtual reconstructions. Nevertheless several benefits and deficiencies of procedural modeling and the CityEngine software package appeared during the generation of the reconstruction of the Sanctuary of Diana. The main shapes were done using the CGA rules and the high detailed parts were modeled in the traditional modeling software SketchUp and they were subsequently imported in the CityEngine project. This procedure enables to create complex 3D model of buildings. The advantage of using this method is the possibility of parameter-based modeling which allows to define parameters (dimensions, level of detail, type of shapes) that can be changed any time. Parameters together with the proportional described relations of the temple enable to modify the parameters without distorting of the entire model. This leads to another benefit of the procedural modeling which is the option of testing several hypotheses of the model by changing certain parameters and attribute. Even though procedural modeling is a powerful technique for the 3D model generation, certain level of the programming knowledge needs to be achieved in order to create the models. It has to be admitted that another disadvantage of procedural modeling remains in the limitation of the level of the shape complexity of the created models. This is caused by the fact that procedural modeling was mainly designed for modeling of cities, thus regular shapes of buildings. Fortunately these limitations did not affect the generation of the Diana Temple because the shapes of the temple are mostly regular. The major deficiency of the CityEngine lies in the inability of creating annotation, labels, and legends. These features are even more important for the explanations of thematic visualizations like assigning the the values of reliability to the corresponding color scheme. Since there is no sophisticated way of how to insert legend or explanatory labels into the 3D scenes, these elements were projected as a textures of flat 2D shapes lying on the ground. This method has to be considered only as a temporary because the labels and legends are legible almost only from the orthogonal direction. Finally, when comparing the workflow of the creation of the model of the Diana temple to the procedures of the recent studies outlined in section 2.1, all the methods do not show any significant differences. Regarding the labelling issue, none of the recent studies dealt with the need of displaying any labels or legends.

Resulting 3D reconstruction

One of the goals of this thesis was to create a virtual reconstruction defined as model that visually recovers a building at a given moment in the past from the available evidence and studies carried out by archaeologists and other experts (Lopez-Menchero et al., 2011). The final reconstruction of the Sanctuary of Diana have many forms in order to visualize the temple reconstruction and the uncertainty. However all the reconstructions are depicting the temple in the third stage of its construction process. The visualizations are accompanied with three levels of details which helps user to understand the reconstruction correctly and to avoid the misunderstanding caused by photo-realistic

modeling. Since providing only the high LoD with textures might lead to an impression that the model appearance is as certain as photograph. This was one of the challenges related to fulfilling all the requirements resulting from the London Charter and the Seville Charter. Because of the Charters demands, the uncertainty visualizations were created and they are described later in this chapter. Furthermore, the web page referring to the created projects has been created containing needed documentation about the temple generation: description of the modeling methods, characteristics of archaeological data, or approaches used for the calculation and visualization uncertainty. Hence the user can get overview about the entire reconstruction process. Publishing the documentation is not common among the recent studies although it is strictly highlighted in the London Charter and the Seville Charter. Contrary to this thesis, recent projects are not using CityEngine Web Viewer tool to publish their models. It is usual that when the models are finished in the CityEngine, they are exported to another rendering software; for instance project Rome Reborn 2.0 (Dylla et al., 2008) was finally rendered in Mental Ray, or Pixar's RenderMan software was used in case of the modeling Mayan buildings in Xkipché (Müller et al., 2006a). Rendering softwares are used to gain realistically looking models by using different types of light settings (such as ambient light) to simulate the exterior lightening.

Handling the calculation of the uncertainty

Uncertainty of the Diana Temple was calculated using fuzzy logic theory. The method for quantification the uncertainty, which is defined in section 3.4.1, is a combination of the approaches of the recent studies delineated in section 2.2. These techniques were modified in order to fit the case study of the Sanctuary of Diana. Thus the resulting method includes reliability components based on studies of Niccolucci et al. (2004), and studies of Tepavčević and Stojaković (2013). Next, the final reliability was calculated by the minimum function according Niccolucci et al. The main disadvantage of this method is the dependency of the reliability value on the order of adding new objects to the model which is caused by the minimum function. On the other hand, the description of the modeling as a stepwise process perfectly suits the environment of procedural modeling (Niccolucci et al., 2004). Other benefit is that the nature of available archaeological data suits the chosen method. Last advantage lies in the possibility of implementation the formula of the minimum function, needed for the calculation of the reliability value, directly in the CityEngine rule file. So the entire model keeps the possibility of interactively changing the reliability values of individual details and the related appearance of the temple. According to the available data, 21 steps of the generation were defined. However textures were not taken in account in the calculations because of the lack of the archaeological evidence.

Visualizing the uncertainty

As it was stated in section 2.3 there are many ways of visualizing the uncertainty. Within this thesis advantages of Green-Yellow-Red scheme and Opaque-Transparent scheme were used to express the exact reliability values. There are three types of uncertainty depictions: The first is indicating uncertainty of individual parts of the model. Secondly, the side by side method was used to highlight the amount of possible hypotheses and it shows the reliability of the model as a whole. Then, thanks to the benefits of fast procedural modeling, sequence of the temple was created. This sequence presents 21 steps of the temple generation where new details are gradually added to these models. Simultaneously, the degree of reliability is expressed using the color schemes, thus the user can get familiar with the gradual increase of uncertainty due to increasing amount of added details. Finally, interactivity was added to the models which consists of obtaining further uncertainty information after clicking on the desired object/temple. Some of the recent studies used similar approaches regarding the color schemes, side by side visualizations, or the interactivity. Displaying of multiple models was directly suggested as an effective way of visualizing the uncertainty in the CityEngine (Van Gool et al.,

2013). However some projects implemented further methods of how to communicate the uncertainty; for instance Strothotte et al. (1999) used different line sketchiness and saturations corresponding to the degree of uncertainty. Furthermore, other authors suggest using glyphs or mapping the uncertainty to the textures (MacEachren et al., 2005; Pang et al., 1997; Sifniotis, 2012). Finally, some studies focus on the animation techniques but these were mostly used for expressing the temporal uncertainty (Zuk et al., 2005).

6 CONCLUSION

The final part of this thesis leads to the conclusion of the methods and results of the 3D reconstruction of the Sanctuary of Diana at Nemi, in Italy. It also concludes the choice of the approaches used for the calculation of the uncertainty and methods of the uncertainty visualizations. In the beginning it should be highlighted that completing this thesis has been challenging for me as a student of cartography since it is a multidisciplinary project that requires knowledge of archaeology, Roman architecture, procedural modeling and mathematics.

Decision whether procedural modeling technique is suitable approach for the generation of the model of the Sanctuary of Diana is complex. It would not be efficient if the wanted result of the reconstruction was only the 3D model without any further requirements or uncertainty visualizations. However in case of visualizing the uncertainty, procedural modeling offers suitable and effective solution. Moreover, CityEngine is a great tool for testing archaeological hypotheses because of the parameter-based modeling environment. Thus using procedural modeling approach in order to create 3D model showing different hypotheses and uncertainty values is time saving and highly effective. Also the environment of the CityEngine is user friendly and easy to manage. Finally, it appeared that the methods of the implementation of the procedural modeling techniques in this thesis are very similar to the methods of the recent studies. This unity indicates that usage of the procedural modeling is very clear and it has natural workflow.

As it was stated in previous chapter discussing the results, method for quantifying the uncertainty using fuzzy method and minimum theory has several advantages and disadvantages. To avoid the disadvantage such as the dependency of the final reliability result on the order of the added details, completely different methods would have to be used. For instance, Sifniotis (2012) suggests several approaches based on the possibility theory, transferable belief model, and generalised Bayesian theorem. These methods are more complex so they could bring a solution for the deficiency of fuzzy logic stated earlier. However they require very detail archaeological information and knowledge about different possible hypotheses, for instance, if the excavated building was a temple or a residential house, or whether the found room was used as a kitchen or a bedroom. Such hypotheses are not compatible with the study case of the Sanctuary of Diana in Nemi, hence the fuzzy logic and minimum function were better to use. Also, the formulas suggested by Sifniotis might be too complex for the calculation of the reliability values directly in the CityEngine unlike the simple calculation of the minimum function. In conclusion, despite the deficiencies of fuzzy logic approach, it was appropriate to use this method for the reconstruction of the Diana Temple. Since the nature of archaeological information suits the selected approach and moreover, the calculation could be encoded directly in the CGA rule files.

Regarding the available archaeological knowledge about the Diana temple, the level of detail of the uncertainty, which equals 21 steps of the temple generation, is reasonable. This amount should correspond the the purpose of the uncertainty expression and to the quantity of archaeological data. Quantity and quality of the data is also related to the amount of details included in the calculations, for instance, textures or different levels of details. According to the London Charter, 3D reconstruction should always cover all phases of the building's construction (Lopez-Menchero et al., 2011). However reconstruction of all three stages of the Diana temple and expression of the related temporal uncertainty would be out of the scope of the master thesis, that's why only the period of the third phase was modeled.

The methods used for visualizing the uncertainty should be selected regarding the characteristics of the CityEngine. CityEngine does not allow any modification of edge's saturation or sketchiness. Neither implementation of glyphs or modifying texture would be possible. Contrary, CityEngine offers great advantage in fast creation of multiple interactive models presenting several hypotheses. It may become powerful tool for the archaeological virtual reconstruction regarding the demands stated in the London and Sevilla Charter. Unfortunately the visual differences of the individual temples of the side by side visualizations are too small. Side by side approach would be more effective with easily distinguishable set of models but this was not possible with the available data of the case study of the Sanctuary of Diana. The only disadvantage of the CityEngine is the impossibility of inserting legends or explanatory labels. Next, the CityEngine Web Viewer enables unique presentation of 3D models directly in the web browser without no need of additional plugins. Furthermore, comparison tools of the Web Viewer give great opportunity to the user to compare between different models and thus correctly understand the available knowledge of the reconstruction. Finally, thanks to the possibility of different layers and comparison tool, audience can get their preferred and popular photo-realistically looking model and simultaneously they can easily switch the display to the veridical model preferred by archaeologists. Finally, the rendering quality of the results gained through the CityEngine Web Viewer is almost identical with the results of the recent studies which were rendered in additional softwares. Thus it has been proved that using of this additional rendering step is not necessary for the scientific purposes.

The challenges of this thesis indicated in section 1.4 were: generation of the 3D model using procedural modeling techniques, expressing the uncertainty of archaeological data using fuzzy logic theory, handling different visualization renderings related to the level of details and model's uncertainty, and lastly publishing the generated 3D models on-line. To conclude this thesis, it can be claimed that all the above mentioned challenges were successfully accomplished.

ACKNOWLEDGEMENT

During my studies of the international program Master of Science in Cartography I have increased my knowledge of cartography and other related fields, I also met many interesting and smart people from all over the world, and I have dozen of amazing memories that I will never forget. Thus I would like to firstly thank my parents for the possibility of studying this master program because without them it would not be possible. Next, I would like to thank my whole family and friends for their never ending support during my studies.

I would like to thank the coordinator of our master program Dipl. -Ing. Stefan Peters from TUM who was always helpful and supportive during the entire time of our studies. Secondly, I appreciate all the stuff from all three universities involved in this master program: Technische Universität München, Technische Universität Wien, and Technische Universität Dresden. Next, I would like to thank Prof. Georg Gartner whose positive and energetic attitude always motivates me to work hard and stay proud of being cartography student.

Finally, I want to thank my supervisor Dr. -Ing. Holger Kumke from TUM for his valuable advice and leadership during writing this thesis. This cooperation provided me a lot of new experiences and impressions. I would like to also mention my second supervisor Dipl. -Ing. Stefan Peters who always found some time to discuss the unclear parts of my thesis. Last but not least, I would like to thank archaeologist Dr. Francesca Diosono for explanation of the excavations of the Diana Temple and for the patient guidance during the model generation.

September 2014
Mariana Danielová

REFERENCES

- 3D-COFORM. *Welcome to 3D-COFORM*. Accessed 2014-9-12. 2014. URL: <http://www.3d-coform.eu/> (cit. on p. 12).
- Ambrose, G., S. Stone, and P. Harris. *The visual dictionary of architecture*. Ava Publishing SA, 2008 (cit. on pp. 34–36).
- Azhar, S. "Building information modeling (BIM): Trends, benefits, risks, and challenges for the AEC industry". In: *Leadership and Management in Engineering*, vol. 11(3) (2011), pp. 241–252 (cit. on pp. 4, 5).
- Bilalis, N. *Computer Aided Design (CAD)*. Tech. rep. Technical University of Crete, 2000, p. 26 (cit. on p. 4).
- Denard, H. "A new introduction to the London Charter". In: *Paradata and Transparency in Virtual Heritage Digital Research in the Arts and Humanities Series*, vol. (2012), pp. 57–71 (cit. on pp. 1–3, 7).
- Di Angelo, M., P. Ferschin, and G. Paskaleva. "Shape Grammars for Architectural Heritage". In: *International Conference on Architecture and Urban Design*. 2013 (cit. on p. 9).
- Duncan-Jones, R. P. "Length-units in Roman Town Planning: The Pes Monetalis and the Pes Drusianus". In: *Britannia*, vol. 11 (Nov. 1980), pp. 127–133 (cit. on pp. 17, 29).
- Dylla, K., B. Frischer, P. Mueller, A. Ulmer, and S. Haegler. "Rome Reborn 2.0: A Case Study of Virtual City Reconstruction Using Procedural Modeling Techniques". In: *Computer Graphics World*, vol. 16 (2008), p. 25 (cit. on pp. 8, 63).
- Ernststrom, B., D. Hanson, and D. Hill. *The contractors' guide to BIM*. Associated General Contractors of America, 2006, p. 41 (cit. on pp. 4, 5).
- ESRI. *CityEngine*. Accessed 2014-8-20. 2014a. URL: <http://www.esri.com/software/cityengine> (cit. on p. 21).
- *Procedural Pompeii*. Accessed 2014-9-21. 2014b. URL: <http://www.esri.com/software/cityengine/industries/procedural-pompeii> (cit. on pp. 8, 49).
- Esri. *3D on ArcGIS Online*. Accessed 2014-9-8. 2012. URL: <http://blogs.esri.com/esri/arcgis/2012/09/21/3d-arcgis-online-cityengine/> (cit. on p. 58).
- *CityEngine Help*. Accessed 2014-7-10. 2014. URL: <http://cehelp.esri.com/help/index.jsp?topic=/com.procedural.cityengine.help/html/manual/cga/basics/toc.html> (cit. on pp. 32, 46, 47, 59).
- Ferschin, P., M. D. Angelo, and G. Paskaleva. "Parametric Balinese Rumah Procedural Modeling of Traditional Balinese Architecture". In: vol. (2013), pp. 199–206 (cit. on p. 9).
- Frischer, B., F. Niccolucci, N. Ryan, and J. A. Barceló. "From CVR to CVRO: the past, present, and future of cultural virtual reality". In: *Virtual Archaeology between Scientific Research and Territorial Marketing, proceedings of the VAST EuroConference, Arezzo, Italy*, vol. (2000), pp. 7–18 (cit. on p. 1).
- Gershon, N. "Visualization of an imperfect world". In: *Computer Graphics and Applications, IEEE*, vol. 18(4) (1998), pp. 43–45 (cit. on pp. 11, 14).
- GoogleMaps. *Google Map Maker*. Accessed 2014-4-23. 2014. URL: <https://www.google.com/mapmaker> (cit. on p. 15).
- Griethe, H. and H. Schumann. "The Visualization of Uncertain Data: Methods and Problems." In: *SimVis*. 2006, pp. 143–156 (cit. on pp. 10, 11, 13, 14, 26).
- Haegler, S., P. Müller, and L. Van Gool. "Procedural Modeling for Digital Cultural Heritage". In: *EURASIP Journal on Image and Video Processing*, vol. (2009), pp. 1–11 (cit. on pp. 3–6, 12, 21, 22).

- Harris, C. M. *Dictionary of Architecture and Construction*. McGraw-Hill Professional, 2005, p. 1040 (cit. on pp. 29–32, 36, 39).
- Hermon, S. and F. Niccolucci. “Estimating subjectivity of typologists and typological classification with fuzzy logic”. In: *Archeologia e Calcolatori*, vol. (2002), pp. 217–232 (cit. on pp. 7, 9).
- “A fuzzy logic approach to typology in archaeological research”. In: *The digital Heritage of Archaeology, Athens, Archive of Monuments and Publications*, vol. (2003), pp. 307–310 (cit. on p. 7).
- Howell, I. and B. Batcheler. “Building information modeling two years later—huge potential, some success and several limitations”. In: *The Laiserin Letter*, vol. 22 (2005) (cit. on pp. 4, 5).
- Jokilehto, J. “Definition of cultural heritage: references to documents in history”. In: *ICCROM Working Group 'Heritage and Society'*, vol. (2005), pp. 4–8 (cit. on p. 1).
- Lindenmayer, A. “Developmental systems without cellular interactions, their languages and grammars”. In: *Journal of Theoretical Biology*, vol. 30(3) (1971), pp. 455–484 (cit. on p. 21).
- London Charter Initiative. *The London Charter*. Accessed 2014-5-2. 2009. URL: <http://www.londoncharter.org/> (cit. on pp. 2, 7).
- Lopez-Menchero, V. M., A. Grande, A. Camilo, J. Cela, C. Real, C. Educativo, P. Jose, M. Blanco, and C. D. Isla. “The Principles of the Seville Charter”. In: *CIPA symposium proceedings*, vol. (2011), pp. 2–6 (cit. on pp. 2, 3, 7, 62, 65).
- MacEachren, A. M. and T. Mistrick. “The role of brightness differences in figure-ground: is darker figure?” In: *The Cartographic Journal*, vol. 29(2) (1992), pp. 91–100 (cit. on p. 10).
- MacEachren, A. M., A. Robinson, S. Hopper, S. Gardner, R. Murray, M. Gahegan, and E. Hetzler. “Visualizing geospatial information uncertainty: What we know and what we need to know”. In: *Cartography and Geographic Information Science*, vol. 32(3) (Jan. 2005), pp. 139–160 (cit. on pp. 14, 64).
- Maïm, J., S. Haegler, B. Yersin, P. Mueller, D. Thalmann, and L. Van Gool. “Populating ancient pompeii with crowds of virtual romans”. In: vol. (2007), pp. 109–116 (cit. on pp. 8, 9).
- Mark, R. *Architectural technology up to the scientific revolution: the art and structure of large-scale buildings*. The MIT Press, 1993, p. 252 (cit. on pp. 32, 72).
- Minialoff, R. “Introduction to Computer Aided-Design”. In: *RCI*, vol. (2000), pp. 3–6 (cit. on p. 4).
- Müller, P., T. Vereenoghe, A. Ulmer, and L. Van Gool. “Automatic reconstruction of Roman housing architecture”. In: vol. (2005), pp. 287–297 (cit. on p. 9).
- Müller, P., T. Vereenoghe, P. Wonka, I. Paap, and L. J. Van Gool. “Procedural 3D Reconstruction of Puuc Buildings in Xkipché.” In: *VAST*. 2006a, pp. 139–146 (cit. on pp. 8, 9, 21, 23, 63).
- Müller, P., P. Wonka, S. Haegler, A. Ulmer, and L. Van Gool. “Procedural modeling of buildings”. In: *ACM transactions on graphics*, vol. 25(3) (2006b), pp. 614–623 (cit. on pp. 4, 5, 22).
- Nemi to Nottingham. *Diana Nemorensis: Origins of the Legend*. 2014-9-20. 2014. URL: <http://nemitonottingham.wordpress.com/2013/06/26/diana-nemorensis-origin-of-the-legend/> (cit. on p. 16).
- Niccolucci, F. and S. Hermon. “A fuzzy logic approach to reliability in archaeological virtual reconstruction”. In: *Proc. of the 32nd International Conference on Computer Applications and Quantitative Methods in Archaeology*. 2004, pp. 26–33 (cit. on pp. 6, 9, 10, 20, 24, 63).
- Noghani, J., E. F. Anderson, and F. Liarakis. “Towards a Vitruvian Shape Grammar for Procedurally Generating Classical Roman Architecture”. In: *VAST: International Symposium on Virtual Reality, Archaeology and Intelligent Cultural Heritage-Short and Project Papers*. 2012, pp. 41–44 (cit. on pp. 5, 9, 21, 23).
- Pang, A. T., C. M. Wittenbrink, and S. K. Lodha. “Approaches to uncertainty visualization”. In: *The Visual Computer*, vol. 13(8) (1997), pp. 370–390 (cit. on pp. 11, 12, 14, 64).

- Parish, Y. I. and P. Müller. "Procedural modeling of cities". In: *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*. ACM. 2001, pp. 301–308 (cit. on pp. 5, 21, 22).
- Patterson, R. "What vitruvius said". In: *The Journal of Architecture*, vol. 2(4) (1997), pp. 355–373 (cit. on p. 17).
- Peters, S. and P. Papakosta. *Archeological Cartographic Information System: Demonstrated for the excavation of the Diana sanctuary in Nemi*. Project Documentation Report. Technische Universität München, Fakultät für Bauingenieur- und Vermessungswesen, 2012 (cit. on pp. 15, 16).
- Prusinkiewicz, P., A. Lindenmayer, and J. Hanan. "The algorithmic beauty of plants". In: *The virtual laboratory (USA)*, vol. (1990) (cit. on p. 22).
- Sifniotis, M. "Representing archaeological uncertainty in cultural informatics". PhD thesis. University of Sussex, 2012 (cit. on pp. 10–12, 14, 64, 65).
- Sifniotis, M., B. Jackson, K. Mania, N. Vlassis, P. L. Watten, and M. White. "3D visualization of archaeological uncertainty". In: *Proceedings of the 7th Symposium on Applied Perception in Graphics and Visualization*. ACM. 2010, pp. 162–162 (cit. on p. 24).
- Sifniotis, M., B. Jackson, M. White, K. Mania, and P. Watten. "Visualising uncertainty in archaeological reconstructions: a possibilistic approach". In: *ACM SIGGRAPH 2006 Sketches*. ACM. 2006, p. 160 (cit. on pp. 23, 24).
- Slocum, T. A., R. B. MacMaster, F. C. Kessler, and H. H. Howard. *Thematic cartography and geovisualization, 3rd Edition*. Upper Saddle River, NJ : Pearson Prentice Hall, 2009, pp. 425–439 (cit. on p. 10).
- Smithson, M. and J. Verkuilen. "Fuzzy set theory applications in the social sciences". In: vol. (2006), p. 97 (cit. on p. 6).
- Storemyr, P. *Monitoring and risk assessment of monuments and archaeological sites in the Nemi basin , Colli Albani , Italy*. Project Report. Expert Center for Conservation of Monuments and Sites, 2004 (cit. on p. 15).
- Strothotte, T., M. Masuch, and T. Isenberg. "Visualizing knowledge about virtual reconstructions of ancient architecture". In: *Computer Graphics International, 1999. Proceedings*. IEEE. 1999, pp. 36–43 (cit. on pp. 12, 64).
- Tepavčević, B. and V. Stojaković. "Procedural modeling in architecture based on statistical and fuzzy inference". In: *Automation in Construction*, vol. 35 (2013), pp. 329–337 (cit. on pp. 9, 10, 12, 13, 24, 63).
- The Doric Order. *Illustrated Glossary of Classical Architecture*. Accessed 2014-7-27. 2014. URL: http://www.doric-column.com/glossary_classical_architecture.html/ (cit. on pp. 34–36).
- Trimble. *SketchUp*. Accessed 2014-7-10. 2014. URL: <http://www.sketchup.com/> (cit. on pp. 28, 33).
- Van der Wel, F. J., L. C. Van der Gaag, and B. G. Gorte. "Visual exploration of uncertainty in remote-sensing classification". In: *Computers & Geosciences*, vol. 24(4) (1998), pp. 335–343 (cit. on p. 14).
- Van Gool, L., M. Prasad, and S. Havemann. *Synthesis of 3D Artefacts*. Final Project Report. 3D-COFORM, 2013 (cit. on pp. 12, 13, 63).
- Vitruvius Polio, M. *The Ten Books on Architecture*. Accessed 2014-8-12. 1914. URL: <http://www.gutenberg.org/files/20239/20239-h/29239-h.htm> (cit. on pp. 16, 17, 20, 23, 29–36, 39).
- Watson, B., P. Müller, O. Veryovka, A. Fuller, P. Wonka, and C. Sexton. "Procedural Urban Modeling in Practice". In: *IEEE Computer Graphics and Applications*, vol. 28(3) (2008), pp. 18–26 (cit. on pp. 5, 21–23).

- Wonka, P., M. Wimmer, F. Sillion, and W. Ribarsky. "Instant architecture". In: *ACM transactions on graphics*, vol. 22(3) (2003), pp. 669–677 (cit. on pp. 5, 22, 33).
- Zadeh, L. A. "Fuzzy sets". In: *Information and control*, vol. 8(3) (1965), pp. 338–353 (cit. on p. 6).
- Zimmermann, H. *Fuzzy set theory - and its applications*. Boston: Kluwer, 1991, p. 339 (cit. on p. 6).
- Zuk, T., S. Carpendale, and W. D. Glanzman. "Visualizing temporal uncertainty in 3D virtual reconstructions". In: *Proceedings of the 6th International conference on Virtual Reality, Archaeology and Intelligent Cultural Heritage*. Eurographics Association. 2005, pp. 99–106 (cit. on pp. 13, 64).

APPENDIX

A Values of the parameters used for the generation of the 3D model


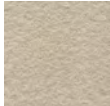


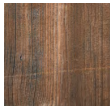

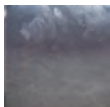

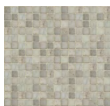
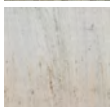
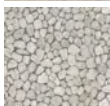
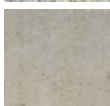
Group	Parameter	Definition (m)	Explanation
model	modelLength	46	measured in ArcMap
	modelWidth	29	measured in ArcMap
podium	podiumHeight	2,8	measured at the excavation site in Nemi
	podiumOffset	0,45	measured in ArcMap
	podiumAssetOffset	0,45	depends on the dimensions of assets
	outsideLength	3,5	measured in ArcMap
	leftCornerLength	5,5	measured in ArcMap
	rightCornerLength	leftCornerLength	equals leftCornerLength
	frontCornerWidth	podiumOffset	equals podiumOffset
steps	stepsLength	4,9	measured in ArcMap
	stepsWidth	15	width of the staircase, measured in ArcMap
	stepsHeight	podiumHeight	same as podiumHeight
	no_of_steps	11	number of steps
	stepDepth	$\text{stepsLength} / \text{no_of_steps}$	based on Vitruvius
	stepHeight	$\text{stepsHeight} / \text{no_of_steps}$	based on Vitruvius
	stepSideWidth	1,3	width of the decorative walls
cellas	leftCellaWidth	7,5	measured in ArcMap
	rightCellaWidth	5,7	measured in ArcMap
	leftCellaLength	10	measured in ArcMap
	mainCellaLength	19,5	measured in ArcMap
	rightCellaLength	12	measured in ArcMap
	mainCellaWidth	$\text{modelWidth} - \text{leftCellaWidth} - \text{rightCellaWidth}$	
walls	wideWallWidth	1,9	measured (estimated) in ArcMap
	narrowWallWidth	1	measured (estimated) in ArcMap
	wallOffset	0,15	offset for the walls
	templeHeight	$\text{columnHeight} + \text{entablatureHeight}$	
columns	bottomColumnDiameter	0,974	measured in ArcMap
	upperColumnDiameter	0,812	based on Vitruvius
	columnBaseHeight	$\text{bottomColumnDiameter} / 2$	based on Vitruvius
	columnHeight	$\text{bottomColumnDiameter} * 8,5$	Diastyle by Vitruvius
	capitalHeight	$\text{bottomColumnDiameter} * 9,5$	Eustyle by Vitruvius
	capitalWidth	$\text{bottomColumnDiameter}$	based on Vitruvius
	capitalWidth	1,1	based on Vitruvius
entablature	entablatureHeight	$\text{architraveHeight} + \text{friezeHeight} + \text{corniceHeight}$	
	architraveHeight	$1/12 * \text{columnHeight}$	based on Vitruvius
	friezeHeight	$3/4 * \text{architraveHeight}$	based on Vitruvius
	corniceHeight	0,359	measured from DWG
	entablatureOffset	0,65	depends on the dimensions of assets
roof	mainRoofAngle	20	units are degrees, based on Mark
	backRoofAngle	2	units are degrees, shed roof
		20	units are degrees, gable roof
	roofTileWidth	0,5	estimated from the evidence
	roofTileLength	0,5	estimated from the evidence
	timberWidth	0,4	no evidence

Continued on next page

Appendix A – continued from previous page

Group	Parameter	Definition (m)	Explanation
	timberHeight	0,3	no evidence
	gapsBetweenTimber	2	no evidence
pediment	pedimentWidth	0,702	assumed from the findings
	pedimentGeisonHeight	corniceHeight	based on Vitruvius
	pedimentCymaHeight	pedimentGeisonHeight * 9/8	based on Vitruvius
	geisonOffset	0,235	depends on the dimensions of assets
	tympanumWidth	0,005	no evidence
doorway	cDoorwayWidth	4	estimated in ArcMap
	cDoorwayHeight	templeHeight * 2/3	no evidence
	sDoorwayWidth	3	no evidence
	sDoorwayHeight	cDoorwayHeight * 4/5	no evidence
window	windowLenght	3,5	no evidence
	windowHeight	4	no evidence
decorations	podiumSDianaWidth	4	no evidence
	podiumSDianaLength	3	no evidence
	podiumSDianaHeight	0,4	no evidence
	statueDianaWidth	3	no evidence
	statueDianaHeight	6	height of the sitting statue
	altarWidth	1,6	no evidence
	altarLength	2,2	no evidence
	altarHeight	1,4	no evidence
	antefixaWidth	0,09	no evidence
	antafixaLength	0,2	no evidence
	antefixaHeight	0,3	no evidence
	antefixaGapLength	roofTileLength - antefixaLength	
	BSPDiameter	5,3	Back Statue Podium diameter
	BSPWidth	0,04	measured in ArcMap
	BSPHeight	0,6	measured at the excavation site in Nemi
	backStatueHeight	4,5	no evidence
	shelterBaseWidth	0,7	no evidence
	shelterBaseLength	4,9	no evidence
	shelterColBaseWidth	1,1	no evidence
	shelterHeight	6	no evidence
	shelterColBaseHeight	0,2	no evidence
	pedimentStatueHeight1	4	no evidence
	pedimentStatueHeight2	3	no evidence
	pedimentStatueHeight3	2,2	no evidence
	pedimentStatueHeight4	1,4	no evidence

B *List of texture used for the reconstruction*

Texture	Type of imitation	Usage	Source
	white plaster	walls, columns	plaintextures.com
	blocks of a stone	podium, entablature, pediment	plaintextures.com
	bronze material	roof tiles	plaintextures.com
	bronze material	roof tiles, statues	plaintextures.com
	wood	timbers	plaintextures.com
	wood	door and window jumbos	plaintextures.com
	glass	window glass	TextureKing
	black mosaic	black mosaic floor	author's work
	white mosaic	white mosaic floor	plaintextures.com
	marble	altar, podium of Diana statue	LuGher Texture
	irregular stones	pavement	plaintextures.com
	stone surface	steps, podium of donated statue	plaintextures.com