

# **3D VISUALIZATION OF HISTORICAL BUILDINGS: METHODS, THEIR UTILITY IN THE TOURISM INDUSTRY AND BEYOND**

Pavel Ugwitz, Lukáš Herman, Zdeněk Stachoň

Masaryk University, Faculty of Science, Department of Geography  
Kotlářská 267/2, 611 37 Brno

E-mail: [ugwitz.pavel@mail.muni.cz](mailto:ugwitz.pavel@mail.muni.cz), [herman.lu@mail.muni.cz](mailto:herman.lu@mail.muni.cz), [14463@mail.muni.cz](mailto:14463@mail.muni.cz)

## **Keywords:**

3D reconstruction, 3D visualization, historical buildings, virtual tourism

## **Abstract:**

What are the reasons and motivation for virtual reconstruction of buildings? What are the possibilities of current-generation 3D visualization in the context of (re)constructing buildings, especially historical monuments, and how can it be utilized in regional development? Such questions are presented and discussed in the article. The following text narrows down the issues from a general/technological overview to focus on the regional development, the tourism industry, and reconstruction of old buildings. Examples of visualization methods are presented and demonstrated with the provided examples. The discussion follows by highlighting the need for collaboration across multiple specializations, considering the untapped potential for utilizing 3D interactive models in the tourism industry, and the ability to observe and collect user behavior in 3D visualization to understand their behavior/interactions better, and to (re)create the 3D user experience to the optimum of their needs and expectations.

## **Introduction**

Significant monuments of human settlements of the past are considered historically valuable or interesting components of the landscape (e.g., for practical utilization in the context of regional history & tourism industry) if they fulfill one or more criteria: the building in question is a remarkable representation of the architectonic style of an era (Davidson, 2017), it is situated in a strategic location with a (once) significant administrative, martial or cultural function (Ching, Jarzombek, Prakash, 2006). To consider such, e.g., with castles, the architectural styles such monuments are (re)built in change through the centuries, and this leads to the castle changing its shape/function through time, or to ruin. With material destruction, however, the historical value of the building is (somewhat) inadvertently lost. Therefore, the question is whether it is possible and/or desirable to create a virtual reconstruction of historical monuments - be it either to capture their current, preserved form, or to at least estimate the shape and form of a building that is no longer there. 3D visualization and documentation of such historical monuments help to keep the history. Moreover, it supports economic growth by stimulating cultural tourism (Koeva, 2016, Scerri et al., 2016).

Through scientific discoveries in the modern era, the scientific fields of archaeology, architecture, and history of material culture have already proposed ways to achieve historical reconstructions. Since the original appearance of a historical monument is not always preserved well, or nearly at all, or is just (wrongly) assumed, some methods of the aforementioned scientific fields do adjust for this: from aerial archaeology (Raczkowski, 2017), negative remains analysis (Angelo, 2017), experimental archaeology (Ascher, 1961), to visual heuristics

based on knowing building construction practices of an era, the missing pieces can, often, be well estimated to provide a probable resemblance of a monument in a given time.

While the aforementioned sciences and methods allow for the extraction of reconstruction data from historical sites even with little or no remains, the standard mediums of reconstruction methods do have limitations - in visuals or adjustability. E.g., physical models are limited by their scale and transportability; if they are to be present in multiple places, multiple copies of the same have to be created. Depictions and imagery are constrained by angle and scale, i.e., the static depiction of a static scale with details or the big picture often being lost. Animations, while digital, are still a set of prerendered frames that offer little/no user interactivity. Enter interactive 3D visualizations, and consider their advantages:

- they can be outlined at first and then worked out in great geometric detail
- 3D structures are transferable across 3D applications and can be worked on using a broad set of production techniques regarding their 3D geometry
- they are (re)editable and customizable (lighting, texture, post-processing)
- they can be made interactive and served through a variety of media, e.g., PCs, smartphones, web interfaces, augmented/mixed/virtual reality

Furthermore, 3D visualization can serve as a basis for producing and distributing the aforementioned static media visualizations (screenshots, fly-through animations, 3D prints), all from a single, easily managed source.

Photorealistic real-time 3D visualizations are now possible due to past technological progress in computer graphics. The technologies have reached a high standard in resolution, 3D model, detail, texturing, and lighting simulation - e.g., through normal mapping, tessellation, physically-based rendering, or ray-tracing (Singer, 2021). Not only this, but the applications purposed for the creation of 3D assets/compositions have evolved their tools to reflect such advancement: while there is more to be done to produce a higher level of fidelity in 3D graphics, 3D tools have streamlined this process, often automating, randomly generating, or visually aiding (Bertolini, 2018) in what would have required manual input in the past. Furthermore, the openness of current-generation 3D real-time graphics engines to scriptability (often using premade classes of C++ or C# languages) allows for custom, tailored interfaces and user experiences.

There are many workflows to achieve a high level of visual fidelity in 3D. For contemporary buildings, BIM (building information modeling (Murphy, McGovern, Pavia, 2009)) is available. BIM - a 3D method intended to replace traditional 2D construction drawings - is five-dimensional: it contains spatial data (3D), with the fourth and fifth dimensions being construction material properties and building modifications through time. Some 3D software, e.g., Tridify BIM tool for Unity, is already streamlined enough to take construction-purposed BIM data and to transform it into a visualization of a building that is often yet to be constructed. For historical buildings, however, we have no such data available; regardless, BIM methodology as such can serve as an inspiration, as similar results are desired to be implementable for historical sites as well - albeit the methods to obtain three-dimensional data of historical sites are different.

Photogrammetry allows for capturing high-detailed 3D meshes of existing (historical) structures. Imaginary, assumed, or otherwise destroyed shapes and structures can then be recreated using either organic modeling (sculpting) or hard surface modeling (Simonds, 2013) using industry-standard 3D software like Blender, Maya, or ZBrush. Generic 3D models can also be reused from an existing 3D model library, e.g., Quixel Megascans, or 3D engine asset

stores (e.g., Unity Store, Unreal Marketplace). The strong suit of a 3D pipeline is that there is no single correct way to (re)create a virtual environment - that is because the 3D techniques and software packages can be used in conjunction with each other. Furthermore, 3D workflows nicely complement and enrich the aforementioned traditional scientific methods. For example, photogrammetry was used in archaeological sites (Jones, Church, 2020), or modeling assumed structures based on known architecture, construction practices, and other (visual) historical sources of an era.

Given the contemporary context of 3D visualization technology and historical visualizations, one has to ask about the utility of such visualizations for the tourism industry. First and foremost, website browsing, mobile browsing/apps, augmented/mixed/virtual reality headsets are all relatively new forms of media that are both becoming increasingly affordable and advanced year by year (Pulli et al., 2007; Dailey-Hebert, Simmons Estes, Choi, 2021). Such mediums can present the user with either the current or the past appearance of a historical site. User viewing can commence either as a supplement to the real visitation of the site or as a substitute/precondition to physically visiting the site (Deng et al., 2020). Provided the user experience and user interface of the virtual application is a good fit for the medium (Weiss et al., 2018) and the targeted types of users (Tomlin, 2018), this can serve to sway the potential visitor to visit the actual site, and/or to enrich the sightseeing experience while physically on site. Therefore, the profitability of such a site can be increased - either indirectly, by converting the potential visitor into an actual future visitor, or directly, by implementing purchases or donations through the virtual application, etc. Disabled or distant potential visitors who would have been otherwise unable to visit a site can also use a virtual visit as a viable alternative - i.e., virtual visitation can represent an improvement in ecology and accessibility.

While the process of creating a virtual representation of a historical site is somewhat universal in terms of 3D graphics (software, supporting reconstruction methods and sources, 3D scene creation pipeline, 3D workflows per specific 3D software), this paper will strive to present such in consideration to the tourism industry, where relevant (user experience, virtual sightseeing commentary).

## **1. Method**

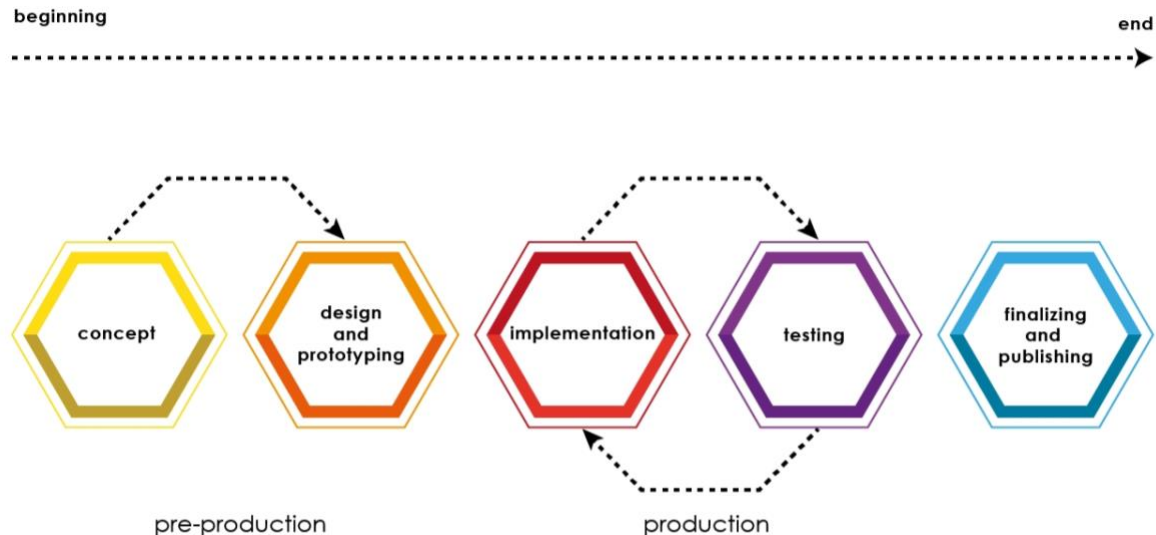
The following process of recreating 3D visualizations of historical buildings, as is to be presented, occurred in the semester of summer 2020 at the Masaryk University, through renovating a master's level college course "3D modelování a vizualizace" (Masaryk University, 2020). The students enrolled in the class were assumed not to have had previous 3D modeling experience; therefore, the semester-long course was intended to cover the fundamentals of 3D modeling. I.e., the students were introduced to what a generic 3D modeling pipeline would look like. To explicate, they were asked to model a castle. Likewise will follow through this article - describing the methods that were taught through the course while merely briefly mentioning the alternatives and advanced approaches.

### **1.1 3D modeling pipeline as a software development pattern**

In the simplest of terms, the process of creating compelling 3D environments can be viewed as following the principles of software development (see Fig. 1). I.e., there is a concept phase that summarizes the ideas that are to be reached. A design/prototyping phase follows to make sure the concepts can be implemented. The implementation phase then carries out the main body of

work - and as it progresses, it can alter back and forth with the testing phase, to ensure the implementation is done to the desired standard. Once such a standard is reached, post-processing and finalizing work is carried out to ensure the results reach a consistent standard. As such, the software is released.

**Figure 1:** generic software development pipeline. It can be applied to 3D modeling as well.



Source: Ferro, Sapio, 2018

In terms of software development models, this can resemble the classic iterative waterfall model, where the whole process goes through from the beginning to the end (McConnell, 1996). More realistically, although the 3D modeling pipeline and its workflows do incorporate some development iterations, e.g., as in an incremental development model, e.g., as in test-based development or SCRUM (Larman, Basili, 2003), since both the software in use and the development/testing of features and visuals can cycle back and forth - especially in case the 3D visualization is a complex one.

## 1.2 Simplified 3D pipeline in use

To outline the structure of the 3D visualization pipeline, we chose to break it down into four segments, all with verifiable and measurable milestones of their own. The students in our course followed such a structure, too:

1. **Conceptual phase.** This phase was meant for the creator to choose a historical castle to model, and to gather as many relevant visualizations and other relevant historical sources as possible, to have a good estimate of what is there to model. If multiple visualizations depicting the same castle were present (be it either through depicting different periods or multiple interpretations of a single period), this did not impose an issue, as there was an inherent explorative component to this - and only by the time of finalizing the concept phase, our students were asked to choose their direction/vision (be it a single existing visualization or a hybrid of a few sources).
2. **Functional prototype.** The purpose here was to create a simplified, gray-boxed, detail-less 3D representation of the visualization to be. That is, only the structures of the terrain and some very primitive man-made structures were extruded, along with placing a first-person movement controller into such a scene. While there were no details, no textures, and very primitive lighting, the purpose of this segment was to establish a 1:1 scale of

the visualization to be, and to be able to walk around the place to ensure all intended areas are connected, accessible, and correctly scaled.

3. **3D modeling.** In this stage, the simplified models need detailing, texturing, and adding other environmental details. This, along with terrain tweaks and adding lighting to the scene, does resemble the majority of the actual work done on a 3D project.
4. **Finalizing and bug fixing.** The 3D scenes created in the previous step were then evaluated by a third party (in this case, student works were evaluated by a teacher; a similar dynamic would ensure similarly with a graphics designer/director, in a professional environment). The creators of the scene were notified of the main issues found in their visualization (if any) and asked to fix these. Graphical postprocessing was introduced into the 3D scenes, and so were performance-minded optimizations.

This pipeline is presented on a conceptual level. Based on the software used (in our case, just ArcGIS, Unity, and Blender), there can be another operational level pipeline. Regardless, that one is hardly specific, as the use of different software combinations can produce vastly different workflows and interdependencies. Traditional hard-surface 3D software (e.g., Maya, 3DsMax, Blender) has its way of handling things, and so do 3D packages based on sculpting or generative paradigms (e.g. Zbrush, Houdini, WorldMachine). The details production pipeline (e.g., Substance Painter/Designer) introduces its quirks, and so do the 3D engines (Unity, Unreal, CryEngine) with their plugins and affiliates (e.g., MegaScans). Coupled with GIS software, land surveying and photogrammetry, the actual 3D development pipeline one ends up with can be a rather complex one.

In either case, adhering at least to the conceptual structure above can prevent issues with 3D modeling of complex structures/composition later on. E.g., as long as a correct building/space scale and user controller interface are established throughout the functional prototyping phase, the other visual/functional issues one can encounter later on will only be minor ones. Similarly, adding lighting and postprocessing details into a scene only makes sense once after the majority of the scene's geometry has been put into place. In contrast to this, ignoring such 3D scene production principles can lead to wasted efforts, where emerging issues may require redoing a significant portion of the previous body of work.

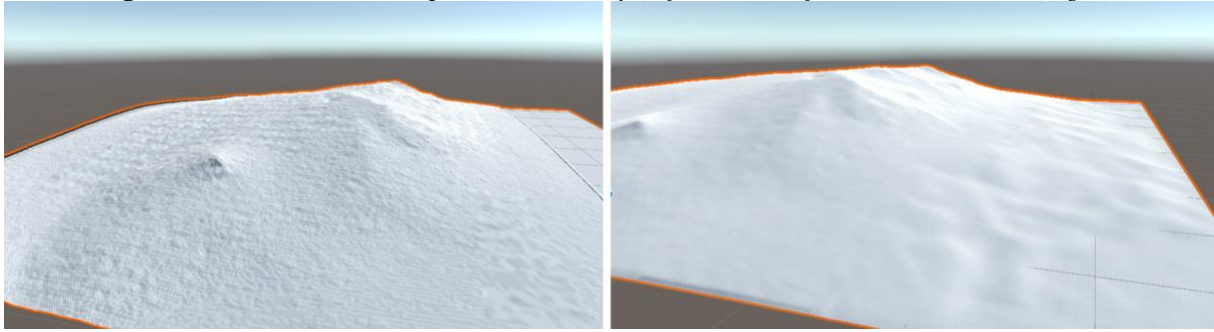
### **1.3 Terrain data acquisition and adjustment**

The 3D terrain models, above which historical monuments were modeled, had been generated from the Digital Terrain Model of the Czech Republic of the 5th generation (DMR 5G). DMR5G was created by the Czech Office for Surveying, Mapping, and Cadastre, based on aerial laser scanning performed by the Military Geographic and Hydrometeorological Office (VGHMÚř).

DMR5G data are available as layer files (.lyr) that are implicitly processed in ArcGIS (in our case version 10.4). In the ArcGIS, the area of interest of the immediate vicinity of the historical monument was extracted (clipped). These extracted Digital Elevation Models (DEMs) were saved as TIFF files).

Upon converting the TIFF DEM into a RAW file, this was then imported into the Unity engine and processed further. As seen in Fig. 2 left, the imported DEM (in Unity terminology, heightmap) did introduce some jitter inaccuracies, given the resolution of the source. Therefore, using the Unity terrain tool (Unity Technologies, 2019a), the jittery terrain was smoothed out a bit to give it natural flow (see Fig. 2 right).

**Figure 2:** *ArcGIS terrain imported into Unity (left). Terrain jitter smoothed out (right).*



*Source: 3D modeling course (Masaryk University, 2020)*

Since the acquired terrain is contemporary, it may not completely resemble its historical shape (due to erosion, sedimentation, man-made activities, etc.) Nevertheless, the Unity terrain tools can be used to bring back the desired shape of the terrain, often in conjunction with placing man-made structures onto the terrain.

Since 3D visualization can have a rather large view distance, spanning up to tens of kilometers, the ArcGIS raster created is recommended to be a large one. While importing the terrain into the 3D engine, this can be further separated into two exclusive terrain objects: a near, highly detailed one, upon which the user walks, and a remote, low-detailed one, which covers the view distance. 3D engines can also use a legacy skybox feature (Unity Technologies, 2019b), but using real terrain data (despite being relatively low resolution) is better to preserve the overall visual fidelity.

Other limitations of the DEMs can occur in case of the terrain being highly vertically differentiated - e.g., cliffsides, large rock formations. Because the DEM is only two-dimensional with a single height value per pixel, it cannot capture nonconvex variance along the vertical axis. In such cases, it is recommended to complement the DEM with 3D vertical models (e.g., obtained through photogrammetry), or to replace it completely with a voxel-based terrain (e.g. Unity Technologies, 2019c). Voxels, however, are yet not very well supported across the current generation of 3D applications, both in 3D engines and GIS software.

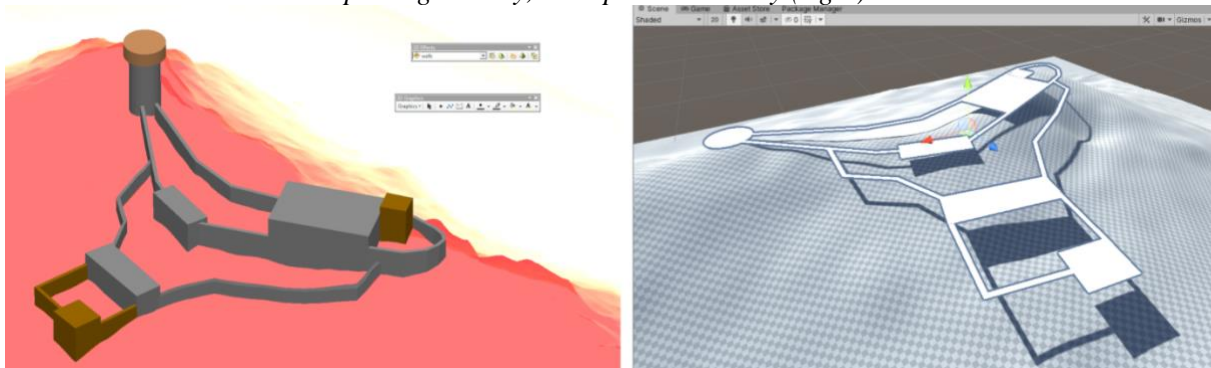
#### **1.4 From acquiring the vector floor plan data to modeling the prototype**

Floor plans of the man-made structures were created manually using appropriately visualized DMR5G data and other supporting data sources (geo-referenced plans, reconstruction drawings, photographs, or potential outcomes from a field survey). Using the ArcGIS extension 3D Analyst and their module ArcScene (ArcGIS, 2021), these shapes were then extruded into the third dimension and saved as so-called Multipatch geometries. Fig. 3 left provides an overview of what the reconstructed castle might look like. These were then also imported into Blender/Unity in the form of Shapefiles.

Since the Blender 3D modeling software has a tight integration with the Unity 3D engine, we used that (simplifying the 3D modeling workflow that was to follow by saving Blender files right into the Unity project, with no importing/exporting needed, as this software did such on their own). Blender, however, did not support the 3D data created in ArcScene (with multipatch geometry), making it effectively so that only 2D floor plans of castle structures were imported and had to be modeled into 3D once again (Fig. 3 right).



**Figure 3:** original extrusion of the basic castle geometry in ArcGIS using ArcScene (left). Only flat plane geometry, as imported in Unity (right).

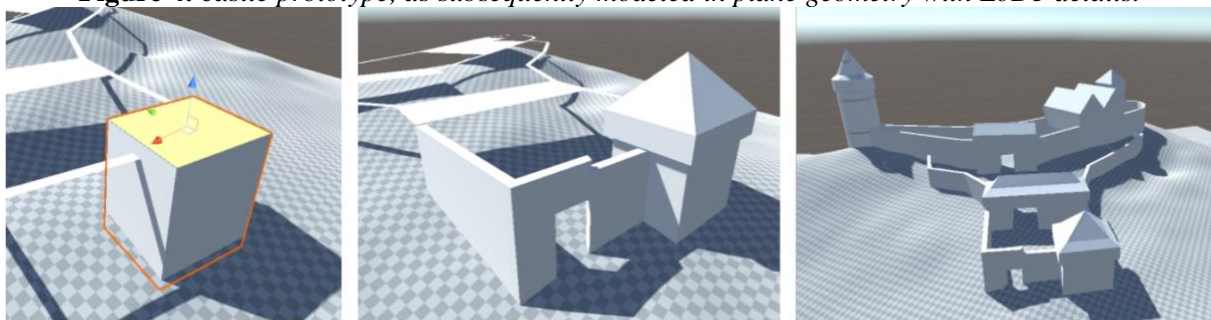


Source: 3D modeling course (Masaryk University, 2020)

Nevertheless, this was not a hindrance for two reasons: one, the 3D geometry to be made required more complexity than ArcScene provided (not just extrusions, but other advanced 3D operations as well), and two, to take this opportunity to teach our 3D modeling class students the principles and basics of the craft.

By modeling the castle from a series of planes with extruded sets of walls and buildings, all with basic doorways and similar physical structures in place, we were able to explain the basics of 3D hard-surface modeling to our students. They tried the different operations available in the vertex/edge/polygon modes and had to think about the desired 3D mesh topology of the objects they were creating (by adding in geometry through extruding beveling, or cutting loops into it; by reducing/optimizing the geometry through deleting polygons and joining vertices; or by using some basic modifiers like mirroring, to aid the modeling process). Such modeling progress is shown in Figure 4.

**Figure 4:** castle prototype, as subsequently modeled in plane geometry with LoD3 details.



Source: 3D modeling course (Masaryk University, 2020)

As seen in Figure 4, such level of detail (based on the level of detail 3 - LoD3, according to the CityGML standard (Kutzner, Chaturvedi, Kolbe, 2020)) was enough for the prototyping phase. We also provided the students with a basic first-person controller (e.g., Halldin, 2018) so that they could walk through their scene prototype and verify their surroundings from the first-person perspective, ensuring that all intended paths were of a correct scale and accessible. In other words, things like oversized fantasy-like towers, oversized stairs, or steep, unwalkable slopes had to be corrected.

While no other interactive scripts were present in our solution (given the already high complexity of the subjects presented to the students), the prototyping phase is also the time to

introduce these. Interactive object functionality, be it animated doors, texts, narration, etc., can be introduced by means of creating a script structure that will carry and deliver their information/interactive content as needed.

Upon finalizing the prototyping phase, we were able to build the first version of an executable, fully walkable 3D visualization application. While our students had different castle assignments to make, they could distribute their prototype builds to their peers to exchange feedback and make adjustments.

## **1.5 Modeling details and texturing**

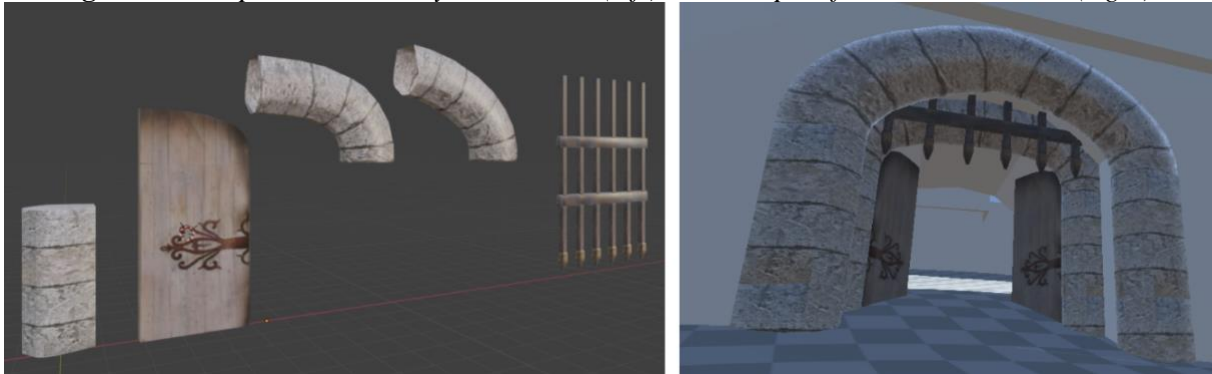
While the process of creating detailed small-scale models is in principle the same as LoD prototyping modeling (the models just get more complex at this point), the issues to be faced and the thinking behind the process are more complex.

First of all, modeling, texturing, and deploying small-scale objects into a scene can be the most time-consuming portion of the whole 3D scene creation pipeline. While being rather simple on the execution level (i.e., a repetitive process of recreating 3D assets repeatedly), this can easily take tens hundreds to thousands of man hours. As such, mismanagement of these efforts can be costly - it is, therefore crucial to have a unifying pipeline and approach in detailing a 3D environment. It is recommended to:

- Have a clear standard of what the 3D assets are to look like. That is, their expected level of detail in both 3D meshing and texturing, a (semi)standardized workflow of their creation, and their overall visual appearance/style. In other words, to have an art direction - in the case of historical visualization, this is likely to follow a realistic art style, but in other cases, e.g., education, something more stylized like conceptual, comic, geometric, or hyper-realistic direction can be used too (Kratt et al., 2014).
- Obtain 3D assets from external sources to save time and resources. For such occasions, there are many 3D marketplaces, some free, some paid. To name just a few, there is the Unity Assets Store, Unreal Marketplace, or Quixel Megascans. Similarly, in our 3D modeling class, we recommended the students share their 3D assets with their peers so that some of the common models would not need creating repeatedly.
- Reuse 3D assets and make some of them modular. It is a common practice in 3D scene detailing to reuse existing assets in multiple instances, to save memory, computing power, and asset creation requirements. Especially, organic, semi-random shapes like rocks can be rotated around their axes, scaled (within reason), and grouped with other similar assets to create unique looking vistas. As for mostly geometric, repetitive, man-made shapes and structures, such can be broken down into modular assets - that is, building blocks, often following a grid structure that can be connected to quickly and accurately create compelling man-made compositions. An example of such a library of modular 3D assets is shown in Fig. 5.



**Figure 5:** *a simple door/archway modular set (left). An example of its use in the scene (right).*

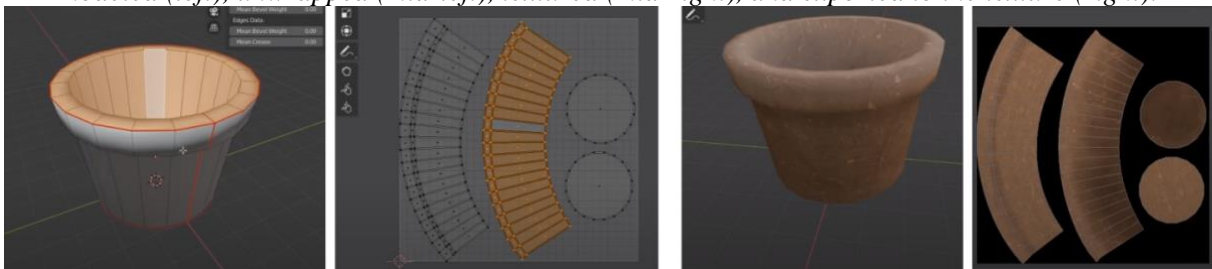


*Source: 3D modeling course (Masaryk University, 2020)*

In our 3D modeling class, we adhered to a simplified 3D asset modeling workflow that consisted of four steps (for a demonstration, see Fig. 6):

1. 3D mesh creation. Simple objects, just like the flower pot in Fig. 6, were advised to be created from imagination, complex objects to be based on a photoreference (ideally as a series of two to three orthogonal pictures, each per axis - i.e., front, side or top).
2. UV unwrapping. That is, projecting the 3D surfaces of a polygonized model onto strips of 2D planes so that they could be textured later on. Since Blender has integrated Smart UV unwrap, i.e., semiautomated functionality to do this (Selin, 2019), we advised our students to use this.
3. Texturing. This was also done inside Blender, thanks to the texture paint and stencil brush functionality (Blender Foundation, 2021). In principle, this step involves obtaining 2D textures from external sources and projecting them onto a 3D object, moving the 3D object around to obtain better projections in place, and finally, saving the projections represented as a bitmap on top of a UV projection as PNG bitmap files. This created the basic texture (also called albedo).
4. Importing the 3D models of assets into Unity - either through an implicit Blender to Unity converter, or by saving the assets as FBX files (a standardized format to maximize compatibility with other potential 3D modeling software).

**Figure 6:** *simplified 3D asset modeling workflow, as demonstrated on a flower pot. An object is modeled (left), unwrapped (mid-left), textured (mid-right), and exported to the texture (right).*



*Source: Abbit, 2020*

While the example workflow described above and shown in Fig. 6 is not exhaustive or producing best-looking 3D assets, it is reasonable enough to be adopted by a 3D modeling beginner, or by a person who strives to produce 3D assets in a reasonable time.

More complex 3D asset creation workflows often also introduce asset sculpting, retopology, baking, and professional-grade texturing (Sairiala, 2015). As illustrated through Fig. 7, a

simplified shape of the desired 3D mesh is created at first; then, it is subdivided and sculpted in high-polygon fine detail, often containing up to tens of millions polygons; a good enough low-polygon retopology of this mesh is then created, and the detail contained in the sculpted mesh is then projected onto it using normal maps. With such baked-in mesh, the texturing process can commence in a professional program like Substance Painter, which assists with procedural texture creation and placement based on the shape and form of the 3D mesh it is being applied on. The stack of specialized textures per 3D asset (i.e, not only albedo for color, but also a normal map for details, a specular map for reflections, an occlusion map for shadow casting, etc.) creates what is known as PBR material (physically-based rendering (Pharr, Jakob, Humphreys, 2016)), which is desired to simulate object material properties across all sorts of lighting settings - provided that scene lighting is crucial, dynamic, or when the purpose of the whole visualization is to reach photorealism (Borg, Paprocki, Madsen, 2014). The price for such a complicated 3D asset creation workflow is the time requirements, which easily span to a tenfold of the simple workflow mentioned above.

Another alternative to 3D asset modeling is their acquisition through photogrammetry (if such props are physically available). With specialized photography equipment (Lachambre, Lagarde, Jover, 2017), the photos can be obtained; then, a specialized photogrammetry software, e.g., Meshroom (Alicevision, 2021) converts these into a 3D mesh; using retopology software like Instant Meshes (Jakob et al., 2015), these can be converted into a mesh that is better suited for editing and 3D engines (all with preserving the normal map and the original texture), corrected to eliminate natural light shadows that may be present in the photos (Unity technologies, 2020), and used.

As for obtaining high-quality textures or materials, there are software packages, e.g., Materialize (Bounding Box software, 2018), that can recreate a whole PBR material based on emulating what would have otherwise required the high-polygon sculpting. While the resulting quality may not be a good fit for all objects and may not always be on par with the sculpting/baking workflow, it sure is a fast way to produce compelling results.

**Figure 7:** *a more complex 3D modeling workflow, as illustrated on a rock formation.*

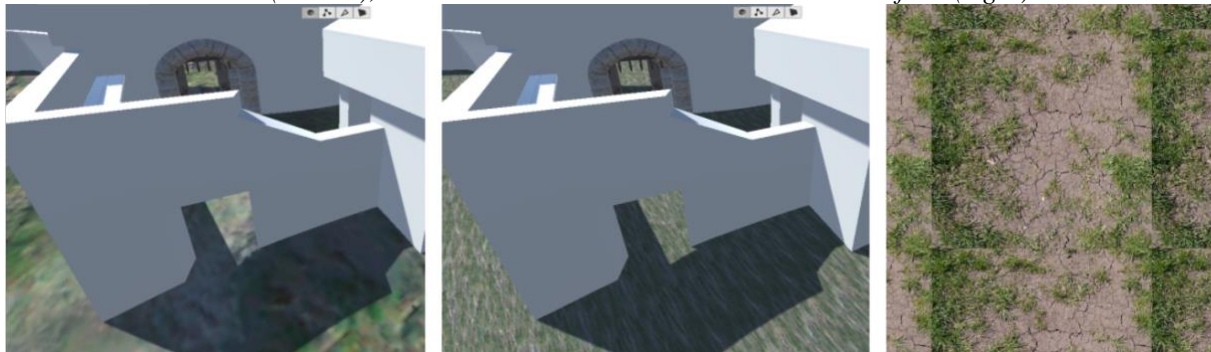


*Source: Waterman, 2014*

With the texturing workflow, one needs to make sure their surfaces do not suffer from issues - that is, the textures are either under scaled, overscaled, skewed in either X or Y axis, misaligned, having visible seams or repetition issues (see Fig. 8; this applies to both 3D textured objects, and texturing the terrain), or being too different compared to textures of other objects (a matter of consistency and art direction). While 3D objects are textured as aforementioned, terrains have to be texture painted as well - the reason for this is the user visiting a 3D reconstruction from a first-person perspective, where a satellite scan of the land would look just too blurry and oversized (e.g., Fig. 8 left). For terrain texturing, 3D engines like Unity have their terrain painting toolsets (for textures, their strength and blending, and oftentimes for low-height vegetation like grass, too); while the height of a 3D in-engine terrain is usually stored in a

heightmap bitmap, the same goes for terrain textures - functionally speaking, these are just more layers.

**Figure 8:** *some issues of texture scaling, use, and alignment. An overscaled texture (left), skewed texture (middle), and non-seamless texture on a seamless surface (right).*



*Source: 3D modeling course (Masaryk University, 2020)*

Completion of a whole 3D scene involves complementing and/or reworking the prototype building by introducing more detailed shapes (unique objects, modular objects, reused objects), terrain heightmap, texturing, vegetation, and clutter. Since man-made structures are oftentimes orderly and geometric, while vegetation and nature form pseudorandom patterns, the contrary of the two can create an interesting dichotomy. By introducing vegetation into a scene, repetitive patterns can be broken. Furthermore, an imperfection in 3D scene composition like texture repetition or 3D model imperfections can be hidden away to an extent. However, artsy, this may sound a good level of integration of all parts present in a scene creating quality environment art (Mon, 2017), which, in turn, makes the user feel comfortable in such a virtual world - i.e. directly improving the immersiveness of the experience (Hudson et al., 2018). What makes or breaks a virtual application is the end feeling of the user/customer. If their impression of a 3D visualization is that a scene is simply “not good enough” (be it because of obsolete or intrusive visuals, cramped space, conflicting user-controller interfaces, etc.), such users would likely not be inclined to visit the real representation of the virtual scene in question. In other words, if a virtual environment is used as a supporting tool to promote tourism, it needs to have a positive conversion rate per potential user/customer (if that is a metric to go by).

This is why the visuals and user interface of a 3D visualization with a real counterpart should cause good impressions in users; this is why they need to be integrated well together. That much can be said for 3D modeling alone; as for user testing for the positivity of such an effect, that is up for discussion.

## **1.6 Scene lighting, post-processing, and optimization**

The purpose of this final stage of the 3D pipeline is two-fold: to make the premodeled 3D scene look presentable considering the context of its use, and to make it run well considering the budget of the hardware it is intended to run on.

Scene lighting is there to accentuate the form and shapes of the composition of the 3D models that were previously put in place. While there surely are dramatic lighting effects like dusk/dawn scenery, photographic magic hours, sun rays (in computer graphics also known as god rays), soft/hard light dichotomies, or reflective, ray-traced scenes (Birn, 2006), which is more of a domain of the film industry, artistic visualizations, or games. When lighting historical

visualizations, a more basic approach can do; from thereon, the question on lighting is whether the scene is situated fully outdoors, or whether it includes indoor space.

Generally speaking, with lighting, it is important to consider light sources. With outdoor spaces, just a single global light source can do - sunlight. This is a light source coming from a distance of infinity, with a global reach across the whole visualization is by default enabled. For a straightforward visualization without a cinematic touch, it is recommended to rotate the sunlight in the vertical axis to achieve an effect of a 3 PM to 5 PM sun, or so. The reason for this is that noon lighting looks bland, not giving the shadows any chance to cast themselves across the existing 3D geometry to accentuate the three-dimensionality of the scene; similarly, early morning or late evening lighting has the issue of casting shadows so long that it is no longer clear what shape originates what shadow and the whole lighting composition gets overlayed by shadows from all sorts of origins (this is especially true when the 3D scene is vertically differentiated - as a castle scene might be).

If a scene contains other light sources, e.g., in indoor spaces with little/no sunlight reach, these are to be included as well. In 3D engines, there are usually two basic kinds of local lights: point lights, which distribute their light in all directions equally, and spotlights, which have a limited angle, in which they cast a light cone (Unity Technologies, 2019d). Such local lights can be adjusted for their limited range and color, among other properties. A deliberate play of light and shadow can give a scene an atmospheric feel, especially in indoor spaces.

Scene postprocessing is a collection of camera filters and effects that can further accentuate the artistic feel of a visualization. Some of them are based on their film or photography counterparts (e.g., color grading, depth of field, vignetting), some others are a domain of computer graphics (e.g., antialiasing, distance fog simulation). While these can be used to hide some imperfections of computer graphics, push the depth of three-dimensionality and photorealism of visualization even further, we do recommend using them sparsely. Not only are they overused in computer graphics media, but they can also be very demanding in terms of performance.

Speaking of performance and scene optimization, it is the beginning of the 3D visualization pipeline (conceptual phase), where the overall requirements for 3D visuals and intended platforms are set. While scene optimization is useful, one does not simply, e.g., convert a lightweight, simplistic scene intended for mobile phones into a photorealistic state of the art virtual reality experience or vice versa. In professional 3D pipelines, the conceptual phase sets a budget (in terms of scene complexity and performance requirements), the finalizing optimizations make sure the budget is not exceeded, along with striving for some reserves. This budget often involves the complexity of 3D geometry (number of polygons simultaneously rendered in a scene), lighting complexity and number of light sources, the size of textures, and the use of postprocessing filters.

While scene optimization can be a complex subject, and while this exceeds both the depth of this article and the topics covered in our 3D modeling course, as taught to our students, adhering to the simple 3D modeling pipeline, as described in this article, ensures performance limits are unlikely to be reached (i.e. a student presenting their semestral work on a standard PC with a dedicated graphics card).



## 2. Results

Each student in our 3D modeling class was tasked to pick a unique castle they would model. Through the concept and prototyping phases of our simple 3D modeling pipeline, the students were advised on their progress, so that the amount of work necessary in one semester's timeframe would be manageable, without complications.

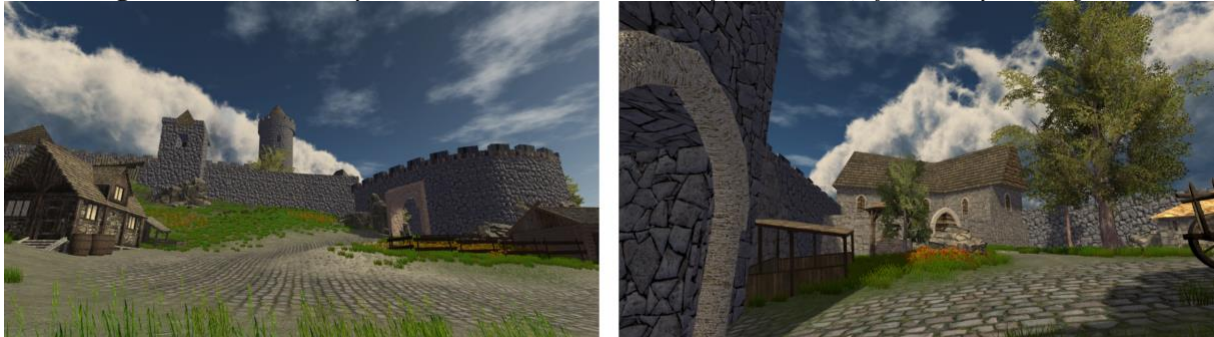
Figures 9-12 present screenshots of the final visualizations of four selected students. The screenshots were taken from the first-person perspective while walking through the virtual environments. This is to illustrate what was made possible in a timespan of one semester; it is also apparent that the students did approach the assignment differently, putting their personalities into the end products.

**Figure 9:** *Čertův hrádek* by Petr Vodička. Outer courtyard (left), inner courtyard (right).



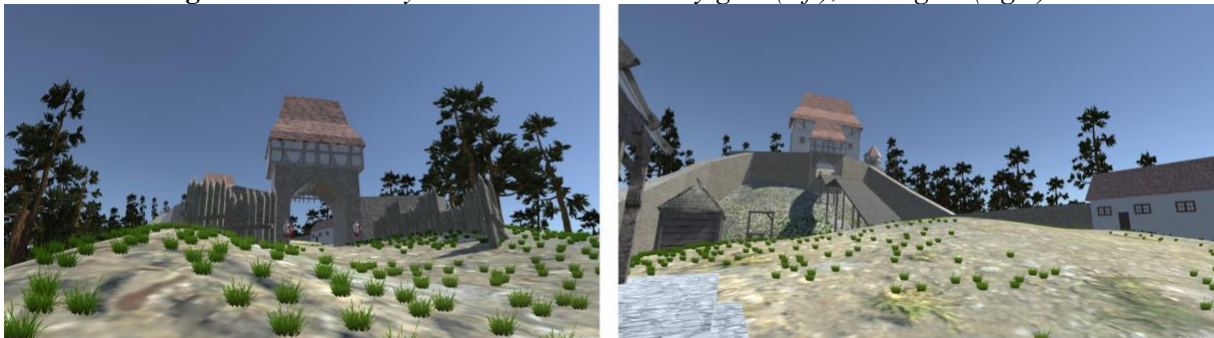
Source: Petr Vodička semestral 3D modeling project (Masaryk University, 2020)

**Figure 10:** *Lichnice*, by Damián Hruban. Outskirts of the castle (left), courtyard (right).



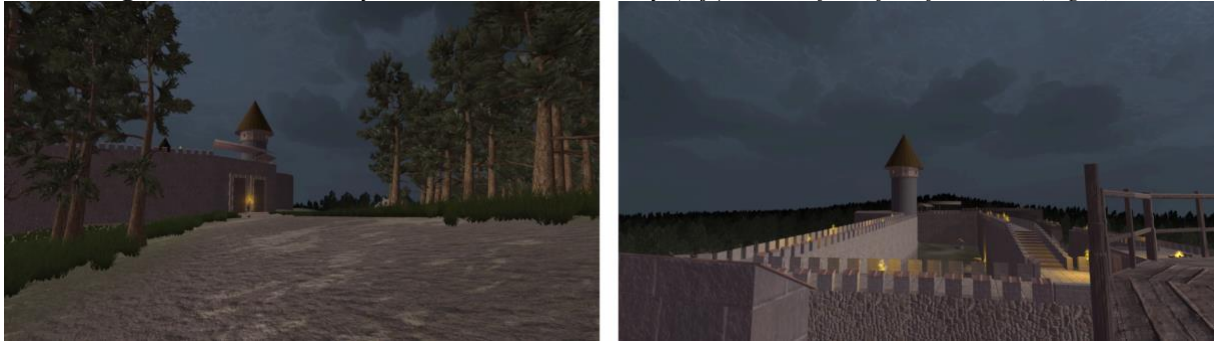
Source: Damián Hruban semestral 3D modeling project (Masaryk University, 2020)

**Figure 11:** *Zlenice* by Rostislav Hořava. Entry gate (left), inner gate (right).



Source: Rostislav Hořava semestral 3D modeling project (Masaryk University, 2020)

**Figure 12:** *Cimburk, by Lukáš Oberreiter. Entry (left), view of the fortifications (right).*



*Source: Lukáš Oberreiter semestral 3D modeling project (Masaryk University, 2020)*

### 3. Discussion and Conclusions

Throughout the article, we argued for the utility of creating 3D visualizations intended to depict historical buildings (be it as a supplementary to physical tourism, or as an alternative to it). Methods for acquiring depictions of the buildings were mentioned - were it methods applicable to fully preserved or completely derelict historical sites. We then considered the 3D methods intended to bring the buildings onto a virtual three-dimensional canvas.

A simplified 3D modeling pipeline for 3D historical visualization was presented in the body of the article. Its viability, in a pedagogical sense, was also verified by our students from a 3D modeling class, who were tasked to create 3D visualizations of selected castles in a timeframe of one semester.

Although the outcomes of this article are limited, it is so due to the complexity of the subject. Thus, the discussion will consider what are the additional steps should the craft of 3D visualization/modeling be pursued further in the context of geographical sciences and the tourism industry.

#### 3.1 Pedagogical perspective

Were the finalized student visualizations presented in the results chapter success or failure? That depends entirely on the perspective of things.

A professional 3D environment artist would say that the visualizations do not meet today's criteria for the level of detail introduced into the scenes and that the modeling has some apparent errors to it. In other words, in today's market, with consumers and enthusiasts being used to a certain level of quality of 3D graphics, the aforementioned visualizations would likely not impress everyone, was it a commercial market.

The creators of the presented visualizations were, however, no professional environment artists with years of experience; these were university students with no previous background in 3D modeling. Therefore, the perspective here is entirely different: to take someone with no understanding of the subject, and to guide them through it, to make them understand the basics and concepts behind it. While the craft behind 3D modeling does indeed take years to master (e.g., in being able to produce a professional level of quality in organic modeling and anatomy), the castle scene seems to be something even a beginner can handle, with reasonable results. To ensure positive results, however, our students were given feedback along the whole

visualization pipeline; and it needs to be stressed that this kind of teaching is suitable for small groups of students, as there is a lot of feedback to give out (our class totaled 8 students).

The students did spend some 20 to 50 hours each on the most consuming phase of the project - i.e., the modeling phase. While a professional may be somewhat faster, such a time budget seems reasonable for a work of depicted quality. The students were also taught not to waste their time by following the recommended 3D visualization pipeline and its software workflows. If they were to decide to follow up on their 3D education, the subject of our 3D modeling class would serve as a good foundation.

It also needs to be noted that teaching future generations of geographers 3D modeling has its relevance. While 2D mapping has had its tradition and utility (Carbonell-Carrera, Jaeger, Shipley, 2018), an increasing number of geographic applications can and do use 3D implementation and visualizations. It is thus more valuable for the scientific field to have skilled experts who can implement or edit 3D applications, as opposed to mere passive observers.

### **3.2 Technological perspective**

Through the 3D visualization pipeline, there are many advanced techniques, software, and plugins to aid or alter the production workflow. Beyond the covered basics, this article only refers to some of the well-known ones, as of 2020. Even though advanced solutions may be out of the scope of this article or an introductory course in 3D modeling, some of the otherwise advanced stuff may find more utility and adoption in the future. For example, there is now procedural art, even in 3D (Matejka et al., 2018), or artificial intelligence-aided texturing and stylization available (Unity Technologies, 2021); it was not so just 10 years ago. Similarly, virtual reality solutions only had appeared on the scene of 3D graphics in the first half of the previous decade, as prototypes; in the second half of the previous decade, production models of virtual reality hardware were made available, with ever increasing availability, user controllability, and overall comfort. Such technologies are here to stay - if not to evolve.

We do not have a universal answer on the implementation of a great 3D application, regardless of the context. Such an answer will always be difficult to grasp, as it depends on many factors out of our current scope: the technology used, the intended level of interactivity of the application, the expected user profile, etc. This is fit for discussion and for future articles - that is, finding the best solutions, graphics pipelines, and workflows in applied settings, or expanding the methodology.

Methodology-wise, user interactivity and user behavior are topics to be considered. While some interactivity depends on the physical hardware used (e.g., an LCD mouse and a keyboard versus a virtual reality headset with controllers), virtual counterparts of physical interactions can be programmed into the 3D application (Ugwitz et al., 2021). In virtual sightseeing applications, implementing some interactivity may aid in having the user behave as intended (e.g., to make them follow along a designed tour path). As for user behavior in general, it can be logged and analyzed; user behavior also depends on the composition of the virtual environment. Evaluating (and adjusting) the spatial composition of environments with known preferences of human users (Wilson, 1999) can improve user enjoyability regardless of the context - be it virtual sightseeing, or something else.



### 3.3 Interdisciplinary perspective

This article takes on the task of presenting and considering 3D modeling, in the context of historical visualizations, to be utilized by the tourism industry. And yet, the authors are no star contributors to everything considering 3D modeling, historical reconstruction, or tourism. In other words, there are more relevant scholars or experts out there. Where exactly are they?

Some 3D modelers, environment artists, and concept artists are self-taught, and some others originate from artistic or technical schools/universities; professionally, they tend to gravitate towards game development studios, film, product design, or architecture. Historical reconstruction using modern technology, e.g., photogrammetry in archaeology, is a niche field. Other historians tend to gravitate towards game studios (Bostal, 2019), or to projects of their own (Univerzita Hradec Králové, 2021). Some geographers explore research niches by depicting historical geospatial data in 3D (Tobiáš, Cajthaml, Krejčí, 2017).

If better visualizations are to be produced, more connections between the people involved in the aforementioned fields/professions need to be established. Modeling a few castles with students is one thing, but there is also untapped potential in professional, applied solutions.

### References

ABBIT, Grant. *Making Simple Game Assets* [online] 2020 [cit 2021-1-31]. Available from: <https://www.youtube.com/watch?v=fEXDylc0Rgg>

ALICEVISION. *Meshroom* [online] 2021 [cit 2021-1-31]. Available from: <https://alicevision.org/#meshroom>

ANGELO, Dante. Histories of a Burnt House. An Archaeology of Negative Spaces and Dispossession. *American Anthropologist* 119(2). 2017. 253-268. <https://doi.org/10.1111/aman.12868>

ARCGIS. ArcMap. *3D Analyst and ArcScene* [online] 2021 [cit 2021-1-31]. Available from: <https://desktop.arcgis.com/en/arcmap/latest/extensions/3d-analyst/3d-analyst-and-arcsene.htm>

ASCHER, Robert. Experimental archeology. *American Anthropologist* 63(4). 2017, 793–816

BERTOLINI, Lucas. *Hands-On Game Development without Coding: Create 2D and 3D games with Visual Scripting in Unity*. Birmingham: Packt, 2018. 404p, ISBN 978-1789538335

BIRN, Jeremy. *Digital Lighting And Rendering*. New Riders Pub, 2006. 416p, ISBN 978-0321316318

BLENDER FOUNDATION. Blender docs. *Texture Paint* [online] 2021 [cit 2021-1-31]. Available from: [https://docs.blender.org/manual/en/latest/sculpt\\_paint/texture\\_paint/index.html](https://docs.blender.org/manual/en/latest/sculpt_paint/texture_paint/index.html)

BORG, Mathias, PAPROCKI, Martin M., MADSEN, Claus B. Perceptual evaluation of photo-realism in real-time 3D augmented reality. In: *International Conference on Computer Graphics Theory and Applications (GRAPP)*, 2014, 1-10

BOSTAL, Martin. Medieval video games as reenactment of the past: a look at Kingdom Come: Deliverance and its historical claim. In: *Del siglo XIX al XXI. Tendencias y debates: XIV Congreso de la Asociación de Historia Contemporánea*, 2019, 380-394. Available from: <http://rua.ua.es/dspace/handle/10045/95829>

BOUNDING BOX SOFTWARE. Materialize [online] 2018 [cit 2021-1-31]. Available from: <http://boundingboxsoftware.com/materialize/>

CARBONELL-CARRERA, Carlos, JAEGER, Allison J., SHIPLEY, Thomas F. 2D Cartography Training: Has the Time Come for a Paradigm Shift? *ISPRS International Journal of Geo-Information* 7(5). 2018. <https://doi.org/10.3390/ijgi7050197>

CHING, Francis, JARZOMBK, Mark M. and PRAKASH, Vikramadi. *A Global History of Architecture*. New York: Wiley, 2006. 864p, ISBN 978-1118981337

DAILEY-HEBERT, Amber, SIMMONS ESTES, Judi, CHOI, Dong Hwa. *Current and Prospective Applications of Virtual Reality in Higher Education*. IGI Global, 2021. <http://doi:10.4018/978-1-7998-4960-5.ch001>

DAVIDSON CRAGOE, Carol. *How To Read Buildings: A crash course in architecture*. Brighton: Ivy Press, 2017. 256p, ISBN 978-0847831128

DENG, Yilin, HAN, Sang-Yun, LI, Jianyi, RONG, Jinjin, FAN, Wenuy, SUN, Tiancong. The design of tourism product CAD three-dimensional modeling system using VR technology. *PLoS ONE* 15(12). 2020. <https://doi.org/10.1371/journal.pone.0244205>

FERRO, Lauren, SAPIO, Francesco. *Unity 2017 2D Game Development Projects*. Packt Publishing, 2018. 338p. ISBN: 978-1786460271

HALLDIN, M. Ludiq. *First Person Controller* [online] 2018 [cit 2021-1-31]. Available from: <https://support.ludiq.io/en/communities/5/topics/1832-first-person-controller>

HUDSON, Sara, MATSON-BARKAT, Sheila, PALLAMIN, Nico, JEGOU, Guillaume. With or without you? Interaction and immersion in a virtual reality experience. *Journal of Business Research*, 100. 2018. 459-468. <https://doi.org/10.1016/j.jbusres.2018.10.062>

JAKOB, Wenzel, TARINI, Marco, PANOZZO, Daniele, SORKINE-HORNUNG, Olga. Instant field-aligned meshes. *ACM Transactions on Graphics* 34(6). 2015. <https://doi.org/10.1145/2816795.2818078>

JONES, Christine A., CHURCH, Elizabeth. Photogrammetry is for everyone: Structure-from-motion software user experiences in archaeology. *Journal of Archaeological Science: Reports* 30. 2020. <https://doi.org/10.1016/j.jasrep.2020.102261>

KOEVA, Mila N. 3D modelling and interactive web-based visualization of cultural heritage objects. In: *International Archives of the Photogrammetry, Remote Sensing and Spatial*

*Information Sciences*, XLI-B5, 2016, 297-303. <https://doi.org/10.5194/isprs-archives-XLI-B5-297-2016>

KRATT, Julian, EISENKEIL, Ferdinand, PIRK, Sören, SHARF, Andrei, DEUSSEN, Oliver. Non-realistic 3D object stylization. In: *Proceedings of the Workshop on Computational Aesthetics (CAe '14)*, 2014, 67–75. <https://doi.org/10.1145/2630099.2630102>

KUTZNER, Tatjana, CHATURVEDI, Kanishk, KOLBE, Thomas H. CityGML 3.0: New functions open up new applications. *PFG–Journal of Photogrammetry, Remote Sensing and Geoinformation Science*, 88(1). 2020. 43-61. <https://doi.org/10.1007/s41064-020-00095-z>

LACHAMBRE, Sébastien, LAGARDE, Sébastien, JOVER, Cyril. *Photogrammetry Workflow*. Unity Technologies, 2017. 97p. Available from: [https://unity3d.com/files/solutions/photogrammetry/Unity-Photogrammetry-Workflow\\_2017-07\\_v2.pdf](https://unity3d.com/files/solutions/photogrammetry/Unity-Photogrammetry-Workflow_2017-07_v2.pdf)

LARMAN, Craig, BASILI, Victor R. Iterative and incremental developments. A brief history. *Computer* 36(6). 2003. 47-56. <https://doi.org/10.1109/MC.2003.1204375>

MASARYK UNIVERSITY. Information System. *Z8311 3D modelování a vizualizace* [online] 2020 [cit 2021-1-31]. Available from: <https://is.muni.cz/predmet/sci/jaro2020/Z8311>

MATEJKA, Justin, GLUECK, Michael, BRADNER, Erin, HASHEMI, Ali, GROSSMAN, Tovi, FITZMAURICE, George. Dream Lens: Exploration and Visualization of Large-Scale Generative Design Datasets. In: *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*, 2018, 1-12. <https://doi.org/10.1145/3173574.3173943>

MCCONNELL, Steve. *Rapid Development: Taming Wild Software Schedules*. Microsoft Press, 1996. 672p, ISBN 978-1556159008

MON, Sierra. ArtStation Magazine. *Everything You Need to Know to Become a Game Environment Artist* [online] 2017 [cit 2021-1-31]. Available from: <https://magazine.artstation.com/2017/03/game-environment-artist/>

MURPHY, M., MCGOVERN, E., PAVIA, S. Historic building information modelling (HBIM). *Structural Survey* 27(4). 2009. 311-327. <https://doi.org/10.1108/02630800910985108>

PHARR, Matt, JAKOB, Wenzel, HUMPHREYS, Greg. *Physically based rendering: From theory to implementation*. Cambridge: Elsevier, 2016. 1266p. ISBN 978-0128006450

PULLI, Kari, AARNIO, Tomi, MIETTINEN, Ville, ROIMELA, Kimmo, VAARALA, Jani. *Mobile 3D Graphics with OpenGL ES and M3G*. Morgan Kaufmann, 2007. ISBN 978-0-12-373727-4

RACZKOWSKI, Włodzimierz. Aerial Archaeology. In: *Encyclopedia of Global Archaeology*. Springer, 2017. [https://doi.org/10.1007/978-3-319-51726-1\\_1504-2](https://doi.org/10.1007/978-3-319-51726-1_1504-2)

SAIRIALA, Joonas. *PBR workflows in Cycles Render Engine*. Tampere University of Applied Sciences, 2015. Bachelor's thesis

SCERRI, Moira, EDWARDS, Deborah, FOLEY, Carmel. The value of architecture to tourism. In *Proceedings of 26th Annual CAUTHE Conference*, 2016, 1-21

SELIN, Erik. Artistic Render. The definitive tutorial to UV mapping in Blender. [online] 2019 [cit 2021-1-31]. Available from: <https://artisticrender.com/the-definitive-tutorial-to-uv-mapping-in-blender/>

SIMONDS, Ben. Blender Master Class: A Hands-On Guide to Modeling, Sculpting, Materials, and Rendering. No Starch Press, 2013. 288p, ISBN 978-1593274771

SINGER, Graham. *The History of the Modern Graphics Processor* [online] 2021 [cit 2021-1-31]. Available from: <https://www.techspot.com/article/650-history-of-the-gpu/>

TOBIÁŠ, Pavel, CAJTHAML, Jiří, KREJČÍ, Jiří. Rapid reconstruction of historical urban landscape: The surroundings of Czech chateaux and castles. *Journal of Cultural Heritage* 30(2), 2017, 1-9. <https://doi.org/10.1016/j.culher.2017.09.020>

TOMLIN, W. Craig. What's a Persona?. In: UX Optimization. Apress, Berkeley, 2018. ISBN 978-1484238660. [https://doi.org/10.1007/978-1-4842-3867-7\\_2](https://doi.org/10.1007/978-1-4842-3867-7_2)

UGWITZ, Pavel, ŠAŠINKOVÁ, Alžběta, ŠAŠINKA, Čeněk, STACHOŇ, Zdeněk, JUŘÍK, Vojtěch. Toggle Toolkit: A Tool for Conducting Experiments in Unity Virtual Environments. *Behavior Research Methods*. 2021. <https://doi.org/10.3758/s13428-020-01510-4>

UNITY TECHNOLOGIES. Unity documentation. *Terrain* [online] 2019 [cit 2021-1-31]. Available from: <https://docs.unity3d.com/Manual/script-Terrain.html>

UNITY TECHNOLOGIES. Unity documentation. *Skyboxes* [online] 2019 [cit 2021-1-31]. Available from: <https://docs.unity3d.com/Manual/skyboxes.html>

UNITY TECHNOLOGIES. Unity asset store. *Ultimate Terrains - Voxel Terrain Engine* [online] 2019 [cit 2021-1-31]. Available from: <https://assetstore.unity.com/packages/tools/terrain/ultimate-terrains-voxel-terrain-engine-31100>

UNITY TECHNOLOGIES. Unity documentation. *Types of light* [online] 2019 [cit 2021-1-31]. Available from: <https://docs.unity3d.com/Manual/Lighting.html>

UNITY TECHNOLOGIES. Unity asset store. *De-lighting tool* [online] 2020 [cit 2021-1-31]. Available from: <https://assetstore.unity.com/packages/tools/utilities/de-lighting-tool-99583>

UNITY TECHNOLOGIES. *Unity ArtEngine* [online] 2021 [cit 2021-1-31]. Available from: <https://unity.com/products/unity-artengine>

UNIVERZITA HRADEC KRÁLOVÉ. Věnná města českých královen [online] 2021 [cit 2021-1-31]. Available from: <https://www.kralovskavennamesta.cz/index.html>

WATERMAN, Patricia. *Trends in the 3D Art Production Pipeline for Games 2014* [online] 2014 [cit 2021-1-31]. Available from: <https://watermanp.wordpress.com/trends-in-the-3d-art-production-pipeline-2014/>

WEISS, Yannick, HEPPERLE, Daniel, SIESS, Andreas, WOLFEL, Matthias. What User Interface to Use for Virtual Reality? 2D, 3D or Speech—A User Study. In: *2018 International Conference on Cyberworlds (CW)*, Singapore, 2018, 50-57. <https://doi.org/10.1109/CW.2018.00021>

WILSON, John R. Virtual environments applications and applied ergonomics. *Applied Ergonomics*, 30(1). 1999, 3–9. [https://doi.org/10.1016/s0003-6870\(98\)00040-4](https://doi.org/10.1016/s0003-6870(98)00040-4)

### **Acknowledgment**

We would like to thank our students from Masaryk University, Faculty of Science, Department of Geography, who participated in the summer 2020 course “3D modelování a vizualizace”. Namely: Simona Bočková, Rostislav Hořava, Damián Hruban, Pavel Kubíček, Lukáš Oberreiter, Jan Přikryl, Aneta Ryglová and Petr Vodička.

Thanks to your input and consideration, we were able to refine the taught subject, which in turn allowed us to publish it in this form.