Digital Applications in Archaeology and Cultural Heritage 5 (2017) 1–9

Contents lists available at ScienceDirect

Digital Applications in Archaeology and Cultural Heritage

journal homepage: www.elsevier.com/locate/daach

Virtual and augmented reality for rich interaction with cultural heritage sites: A case study from the Roman Theater at Byblos

Georges Younes[a], Rany Kahil[b], Mayssa Jallad[c], Daniel Asmar[a,*], Imad Elhajj[a], George Turkiyyah[b], Howayda Al-Harithy[c]

[a] *VRL Lab, American University of Beirut, Lebanon*
[b] *Computer Science department, American University of Beirut, Lebanon*
[c] *Department of Architecture and Design, American University of Beirut, Lebanon*

ARTICLE INFO

Keywords:
Augmented reality
Virtual reality
Roman theater
Byblos

ABSTRACT

The conservation and valorization of our cultural heritage has become one of the priorities of the international community. Accordingly, many technological applications are being developed with a cultural object as the focal theme of interest. This paper presents a detailed account of the construction of a computerized model of the Roman Theater of Byblos, one of oldest continuously inhabited cities in the world. Central to the paper is a historical study, which yielded the formation of a hypothesis on the original shape and details of the theater, which today retains very little of its original structure. Another major part of the paper is the detailed description of the procedures for the creation of both a virtual and augmented reality application of the Roman theater.

1. Introduction

The full appreciation of cultural heritage sites often presents substantial challenges to casual visitors and tourists. The partially destroyed state of these sites and the lack of contextual clues about the functions of their various spatial elements and significance of their architectural features make it difficult to interpret the sites and get a sense of their relevance and importance.

Virtual and augmented reality technologies promise to alleviate some of these difficulties by providing historically accurate reconstructions and relevant contextual information to allow for a richer and more impactful visitor experience. Users wear 3D stereo viewers and headphones, and optionally hold a control pad, to be immersed in a realistic, interactive, and much enhanced environment. In this immersive world, they can navigate or be guided through a historical site that has been augmented with virtual recreations of missing features and enhanced with narratives about the meanings and roles of its elements.

Both on-site and off-site experiences can be enriched in this way. On-site, a video stream of the actual (often partially destroyed) site can be overlaid with a virtual model that augments the heritage site to reveal missing historical elements. Off-site, a 3D replica of the actual site can be overlaid with the virtual model to recreate the historical site. This immersive first-person experience is far more engaging to the general public than a mere walk through the ruins and allows lasting impressions of artifacts, landscapes, and cultural sites to be formed.

In this paper, we report on a system developed for creating such an immersive augmented experience for the Roman Theater at Byblos. Byblos, located along the Mediterranean Sea coast at approximately forty kilometers north of the Lebanese capital Beirut, is one of the oldest continuously inhabited cities of the world. Modern scholars date back its existence to at least the Neolithic Period (around 8000–4000 BCE) (Dunand, 1954). The main characteristic of the city is the superposition of ruins belonging to a succession of civilizations spanning 7000 years of history.

Of particular interest in the Byblos archeological ruins is a Roman Theater, which dates back to the year 218 CE. Little remains of the theater and what still stands is in a poor state, which makes it very difficult for tourists to appreciate its function and its value from a simple visit. The Roman Theater is therefore a prime candidate that could benefit from technological interventions to better highlight to visitors its features, its purpose, and its appearance in the past.

A number of challenges had to be overcome in order to design and implement practical systems to allow seamless and effective user interaction with Augmented Byblos. In this paper, we describe these challenges and the solutions we developed for them. A video of a virtual tour through the augmented site is attached.

The remainder of this paper is structured as follows. Section 2

---

[*] Corresponding author.
E-mail address: da20@aub.edu.lb (D. Asmar).

http://dx.doi.org/10.1016/j.daach.2017.03.002
Received 27 December 2016; Accepted 23 March 2017

Fig. 1. (left) Byblos Roman Theater in its current state, (right) Bacchus mosaic that was found inside the theater.

presents the methods used to generate a 3D computerized model of the existing site, Section 3 details the theoretical analysis leading to a hypothesis of the missing parts of the theater and includes a description of the developed archeological object, Section 4 discusses the development of an augmented reality application while Section 5 details its virtual reality counterpart. The paper concludes with Section 6.

2. Reconstruction of the existing Roman Theater at Byblos

The Byblos Roman Theater was discovered in the late 1920's by Archeologist Dunand, (1954). Today, as shown in Fig. 1 (left), the theater consists of the following: of the original cavea (auditorium/seating area), only the first five rows remain with seven sectors divided by stairs. The orchestra's semi-circular ground was covered with a simple white mosaic with a black border. At the center of the orchestra, there used to be a finely detailed mosaic representation of the Roman god Bacchus (also known as the Greek god Dionysus), the god of the theater. The mosaic is now in the Lebanese National Museum and is shown in Fig. 1 (right). The raised floor of the pulpitum (stage) has five niches adorning its side, each defined by two Corinthian type columns and a pediment. A stone altar that was found at the foot of the seating area opposite to the center of the stage was probably used prior to any performance in a religious ritual to obtain the blessing of the gods. Postholes along the edge of the first row of stone seats in which wooden posts could have been set: perhaps for an awning to shade the audience, or a low barrier to separate the audience from the orchestra area.

2.1. 3D computer modeling of existing structure

To generate a 3D computer model of the theater, three steps are required; namely, (1) extracting a 3D point cloud of the object, (2) fitting geometric primitives to the point cloud, and (3) texturing.

2.1.1. 3D point cloud generation

Traditionally, a 3D point cloud of a scene is extracted using 3D lasers (*e.g.*, Leica[1]); however, such equipment is prohibitively expensive and the mapping requires careful setup of the laser at different locations throughout the mapping. Alternative methods of point cloud extraction use stereo cameras or Kinect[2] systems. Stereo cameras are limited in range and the Kinect cannot be used outdoors because of the noise caused by sunlight. A 3D point cloud of a scene can alternatively be extracted using a single camera, by taking overlapping images of the same scene and estimating 3D coordinates by triangulating matched features across images. The camera motion (rotation and translation) between different viewpoints is obtained using projective geometry techniques. This entire process is known as Structure from Motion (SfM).

SfM techniques suffer from several limitations. First, SfM methods produce dense point clouds at the expense of processing time and hence are considered offline methods. Furthermore, the solutions for structure are ambiguous up to an unknown scale. Finally, as any other visual-based technique, their success relies on a critical number of detected features to succeed. In other words, SfM fails when looking at a texture-less scene such as a uniformly colored wall.

For our application, these limitations were not critical. First, time was not a factor in our application and SfM was left to iterate for several days to extract the point cloud. Second, outdoor environments (such as in archeological sites) are rich in features, which are distinctive and ideally suited for SfM solutions.

2.1.2. Mesh generation and texturing

Given a dense 3D point cloud, the standard workflow for generating the three dimensional models to be used in a virtual environment consists of (1) connecting the points to generate a three-dimensional triangle mesh that approximates the geometry. This is done by a spatial Poisson surface reconstruction algorithm that doesn't rely on heuristic spatial partitioning or blending and as a result is highly resilient to data noise; (2) decimating the dense triangular mesh to reduce it to a manageable size, from tens of millions of triangles to a few thousand representative ones preserving the original topology and a good approximation to the original geometry; (3) applying texture maps to give the geometric model a realistic visual appearance. This is conveniently done here because the texture atlas can be readily produced from the images acquired for the 3D reconstruction; (4) applying normal maps which control light reflections to allow the low polygon meshes to recover the subtle details needed for realism without an increase in polygon count.

We will not discuss the details of the algorithms of this workflow but only note that the main practical concern in the model reconstruction is to balance two conflicting demands: one is for increased realism which requires large and expensive meshes, and the other is for small meshes that can be handled at interactive rates for stereo display. Balancing these two concerns requires some trial and error.

3. Construction of the hypothetical historical model

Analyzing the existing structure and synthesizing a hypothetical model of the missing structure proved to be challenging due to the scarcity of the remains. In addition, the theater's displacement from its original position meant the need to be critical of the disposition of the stones and the preciseness of the relocation procedure, thus the challenge to align the theater with a typical, perfect, geometrical model. Nevertheless, to surmount these challenges, multiple sources were used to inform the process of reconstruction and approached the specific geometry of the remains in a critical manner. The Vitruvian model was

---

[1] http://lasers.leica-geosystems.com.

[2] www.xbox.com/en-GB/xbox-one/accessories/kinect-for-xbox-one.

used as a geometrical base for our reconstruction and compatible regional examples were found amongst the site's contemporaries, in particular, the Odeon at Aphrodisias (Turkey, third Century BC) and Aspendus (Turkey, 161–169 CE). In what follows, the analysis is presented, leading to the development of the hypothetical model, starting with a comparison to Roman and Greek theaters, followed by the fitting of a geometrical model to the existing structure, ending with the hypothesized model used in the work.

3.1. The Byblos Theater: a comparison with the Vitruvian Greek and Roman theoretical models

Although Jidejian (1971) describes the Roman Theater as having Greek characteristics such as its 'seven sectors of steps', it was built in 218 CE, during the Roman era in Lebanon (64 BCE to 600 CE) (Jidejian, 1971), 282 years from the Greek era (333 BCE to 64 BCE). It also has a clearly raised stage with Roman style niches and columns and a Bacchus mosaic characteristic of the Roman times. All these indications have led us to project onto the Byblos Theater a Roman Vitruvian model with several regional variations.

Finding the adequate construction lines to build a 3d model of the theater's *frons scaenae* proved to be a difficult task. The imprecision measured from the 3d model is most likely due to the displacement of the theater in the 1920s. Greek Vitruvian construction lines and Roman Vitruvian construction lines both indicate steps that are at thirty degrees rotation from each other, the former built by three rotating squares and the latter by four rotating triangles. Both models superposed with the precise 3d geometry of the site were not compatible. The site was displaced in such a bad manner that it did not fit a true Vitruvian model. Although it might seem to be most fitting in theory, the Vitruvian Greek construction lines (Fig. 2) are inadequate for the reconstruction of the Byblos theater, due to the angle discrepancies, the positioning of the stage and the side cunei, which go beyond the site's stage diameter in the theoretical Greek Vitruvian model.

3.2. Developing a hypothesis for the Byblos Theater

This section describes the process by which a hypothesis was generated for the Byblos Theater, including details regarding the cavea, scaenea Frons, pulpitium, niches, and columns and doors.

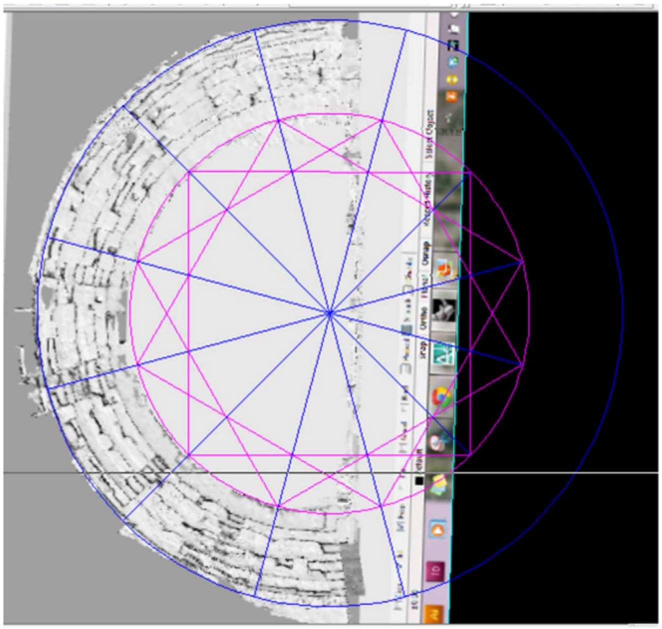


Fig. 2. Byblos Theater superposed with Vitruvian Greek Theater construction lines.

3.2.1. Cavea (audience)

Although The Byblos Theater follows the steps' disposition of classical Vitruvian Greek theaters with seven sectors, it seems to have steps that are less than thirty degrees radially apart. The closest fit was adapting the steps to a Vitruvian Roman model with five rotating triangles (Fig. 3) instead of four (making for 24° apart sectors). Using the Greek model and adding a fourth square would have resulted in sectors 22.5° apart, which is too little. Had this triangle addition and rotation not been made, the model constructed would have been too shifted from and inconsistent with the site's 'as-is' model. Designer input was necessary to think outside the classical Vitruvian model for the deconstruction of this regional variation. Adding an extra rotating triangle was not uncommon in regional variations of Roman Theaters, as the Aspendus Theater will prove.

The plan of the Aspendus Roman Theater (Fig. 4) is off-center and has the same Greek-inspired radiality of steps as Byblos. The scale of Aspendus is much larger and hence contains an *aditus maximus* and 9 sectors of stairs. Measuring the angles between the *cunei* at Aspendus, we found them to be at 20° apart, which can be constructed using six rotating triangles. It is hence possible the architects of these regional theaters manipulated the rotating construction triangles of the classical Roman Vitruvian model to fit within the site's scale and needs.

"Coins found in the foundations of the Byblos theater date the construction to 218 CE" (Jidejian, 1971), which makes it possible that the Aspendus Theater, built in 169 CE (49 years before) was a regional influence to Byblos.

3.2.2. Scaenea frons (theater wall)

The Aspendus Theater (Fig. 5, left), constructed in Turkey in 161–169 CE, was for us an important reference since the small niches on the sides of the Byblos theater stage are very similar to those on the *frons scaenae* of Aspendus, especially the central broken pediment. This helped us understand the details of the frons scaenae, which complemented the Vitruvian sectional proportions we used.

3.2.3. Pulpitum (stage)

The *pulpitum* was based on the Roman Vitruvian model, equal to twice the diameter of the *orchestra* circle. It is the only remaining part of the Scaenae (Theater building). Today its central part is unpaved, as it was perhaps covered in mosaic. The unpaved area is surrounded by white marble tiles.

3.2.4. Small niches

The Niches at the side of the theater are each framed by two small Corinthian-style columns and a pediment. The central niche has a broken triangular pediment and four small Corinthian-style columns. These niches most likely contained small statues of gods, emphasizing the religious nature of this theater. They were our insight to the composition of the *frons scaenae* wall.

3.2.5. Columns and doors

The positioning of the columns and doors of the frons *scaenae* was deducted from the intersection between the construction lines for the hypothetical model's rotating triangles and the back edge of the *pulpitum*. The columns modeled echoed the small ones at the niches. Doors to the backstage corridor were placed in the center and after each group of four columns, inspired by the Aspendus theater's *frons scaenae*.

3.2.6. Height of theater

The height of the theater was calculated according to the Vitruvian sections as equal to the diameter of the circle of the construction lines.

3.2.7. Top of cavea

The top of the Cavea is defined on the inside of the theater by a colonnade and on the outside by a wall with openings to allow

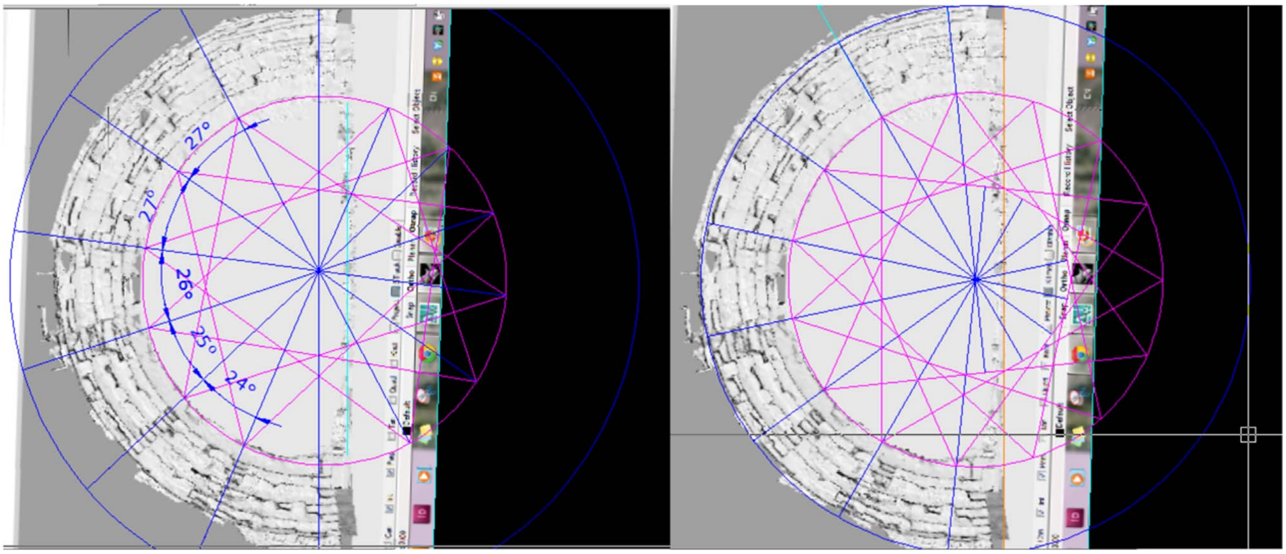


Fig. 3. (left) Byblos Theater superposed with actual construction lines; (right) Byblos Theater superposed with Roman Vitruvian model variation of 5 rotating triangles instead of 4: creating angles 24 degrees apart.

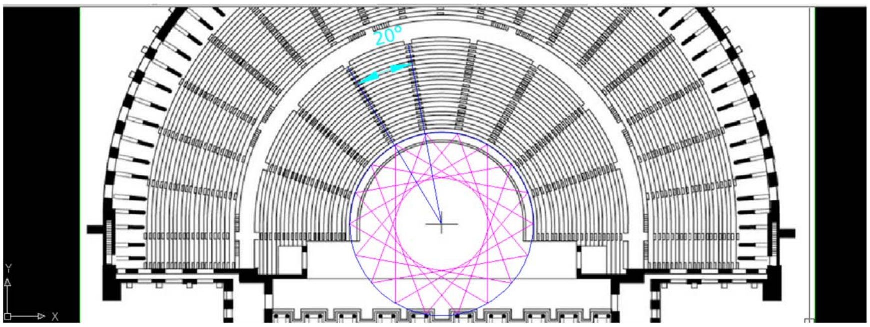


Fig. 4. Aspendus Theater with construction lines: 6 rotating triangles for 20° wedges of cunei.

spectators to enter, inspired by the Aspendus Theater. Its section is also defined by the Vitruvian Roman model's section.

The Final Model is a result of investigative research and comparative analysis of the regional variations that make the Byblos Theater unique. The hypothetical 3d model created is not intended as the “authentic” Byblos Theater, having as only physical evidence the Byblos displaced and pillaged ruins and regional models. The hypothetical model produced is of a Roman theater with 5 rotated triangles for construction lines, a regional variation of the classical “4 rotated triangles” Vitruvian model. Its stairs disposition allows for the position of the stone altar on which incense was burned for Bacchus, the god of Love, Wine and Theater. The *scaenae frons* proposed duplicates the niches found at its *pulpitum*. The proportions of the Vitruvian Roman section were followed. Finally the alignment was respected as best as possible with the 3D reproduction produced, shifting the center of the defining circle, always taking into account the errors provoked by its displacement at its original discovery, with the hopes it will produce an interesting simulation and shed light on the mystery of the Byblos theater ruins.

Fig. 5. (Left) the reconstructive drawing of the Aspendus Theater (copyright © 2004 Thomas G. Hines, Department of Theater, Whitman College). (Right) shows a similar central broken pediment as that of the side niches of the Byblos Theater.

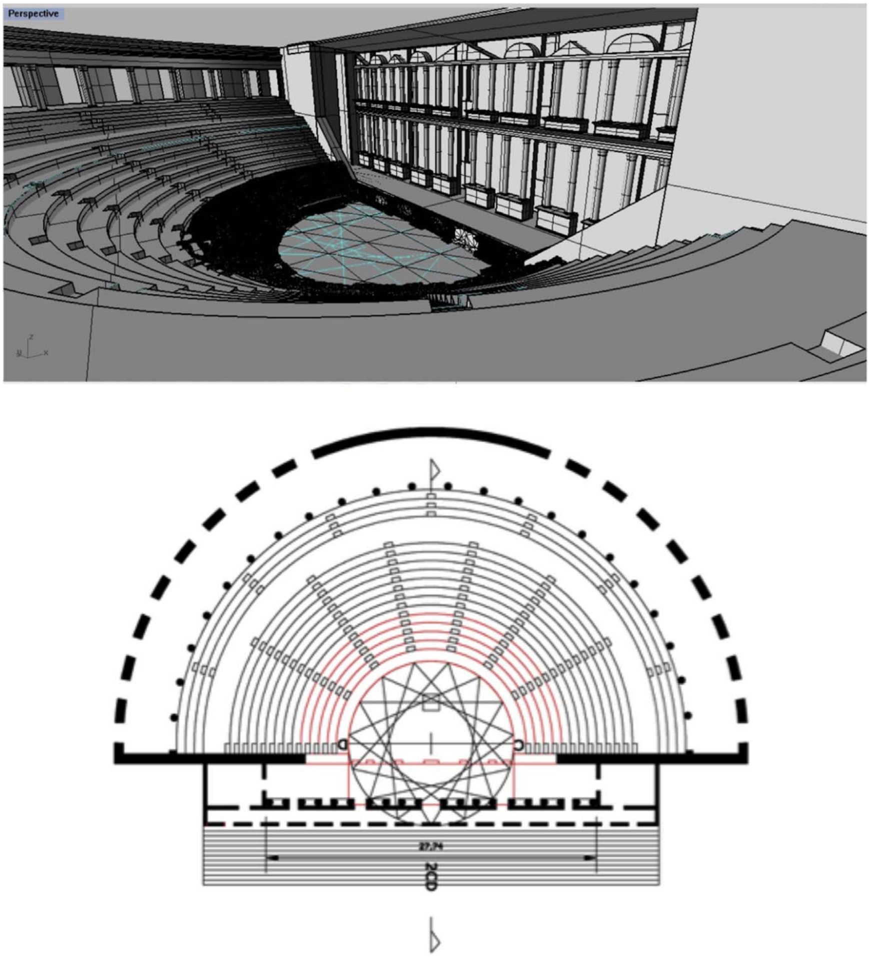


Fig. 6. Final computer model of the augmented Byblos Roman Theater. Perspective view (top) and top view (bottom).

3.3. *Final computer model of the complete Roman Byblos Theater*

The 3d Model was built using the Rhinoceros 3d software[3] as shown in Fig. 6. The "as-is" model presented the necessary evidence for creating the most fitting hypothetical model. The meshing together showed the multiple variations and imprecision the displaced Byblos theater presented compared to the theoretical Vitruvian classical model. The exact reproduction of the remaining ruins was crucial to determine the geometry and proportion of the Byblos Theater. For example, the first regional variation noticed thanks to the as-is model was that the orchestra is off-center, not quite like the roman Vitruvian model. The center of the orchestra was shifted about one meter away from the stage towards the auditorium, which made diameter smaller, altering all the theater dimensions proportionally.

4. Interacting with the augmented model in situ

Augmented reality applications extend and enrich in real-time what tourists observe in a scene; they offer an informative experience by augmenting on top of the existing ruins the hypothesized model.

4.1. *Pose tracking methods*

To achieve a realistic augmentation, the point of view from which the users are observing a scene must be accurately estimated and tracked in order to project to them an augmented object from the correct perspective. Tracking methods can be either infrastructure-based or infrastructure-less.

Infrastructure-based solutions include active systems such as Vicon,[4] or the Optitrack[5] and require altering the archeological/touristic site to add relatively expensive hardware, which require special maintenance and technical care. Infrastructure-based solutions were not considered in this work; instead, we relied on using a camera with specialized software to estimate pose. Over the past three decades many solutions were developed for monocular camera motion estimation, under the title of VO (Visual Odometry) (Scaramuzza and Fraundorfer, 2011), where the camera's pose is incrementally estimated as it explores the scene. Unfortunately, VO tends to drift considerably over time resulting in the scene itself drifting and yielding a poor augmented reality experience. Alternatively, a more robust track can be estimated by building a map of the scene in which one is navigating and localizing oneself inside the map—a technique known as Visual Simultaneous Localization and Mapping (SLAM) (Durrant-Whyte and Bailey, 2006; Younes et al., 2016). During a site visit, the pose-estimating camera can be carried by the user or fixed to part of his/her clothing (such as on a helmet).

4.2. *Parallel tracking and mapping*

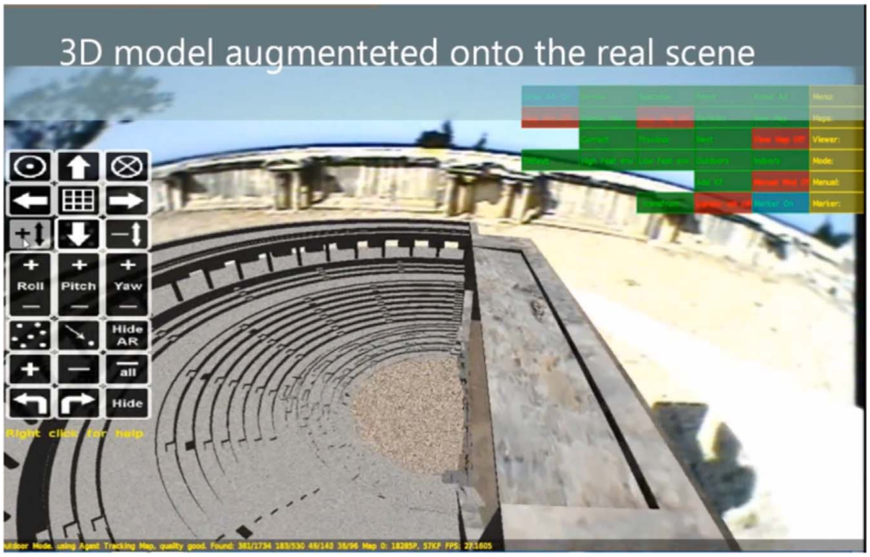
Short for Parallel tracking and mapping, PTAM is an open source Visual SLAM system capable of solving for a camera pose and a map of the observed environment in real-time. It uses automatically extracted

---

[3] https://www.rhino3d.com/.

[4] http://www.vicon.com/.

[5] http://www.optitrack.com/.



```
utdoor Mode, using Agast Tracking Map, quality good. Found: 381/1734 183/530 48/140 38/96 Map 0: 18285P, 57KF FPS: 27.1605
```

Fig. 7. Un-scaled and unanchored virtual object augmented on a live video feed in site.

features from a video feed to estimate the ego-motion of the camera recording the feed. In PTAM the tracking is bootstrapped at startup with an initial map of the scene along with a camera pose. PTAM then proceeds by extracting features at every incoming camera frame and attempts to match them to the map. The correctly matched features are used in an iterative optimization scheme—known as Bundle Adjustment (BA)—to recover the camera pose at the frame. Tracking quality is assessed and if deemed bad, failure recovery methods are invoked; otherwise, a virtual object is augmented on the frame using the camera pose estimate and displayed to the user.

4.3. *Challenges of visual SLAM in AR*

For a successful implementation of Visual SLAM in an AR context, many factors need to be addressed: map scene generation, measurement scale estimation, anchoring of virtual parts to the actual scene, and lighting variations in outdoors environment.

4.3.1. *Map generation*

PTAM achieves real-time performance by selecting only special frames known as keyframes for map generation. Keyframes are distinguished from regular frames using an automatic selection criterion based on significant camera pose change. Fortunately, the first stage of our suggested strategy allows for redundant keyframe addition into the map as it is performed by an expert, who is aware of the system requirements when subjected to deteriorating performance; furthermore, during the scene exploration phase, augmented reality is not performed and hence more resources may be dedicated to the map generation thread in contrast to the tracking only mode where the map generation is disabled.

Another aspect to take care of during the scene exploration phase is the generation of the map's landmarks. The map is made of 3D landmarks triangulated from 2D features. Visual SLAM systems rely on robust data association of features across different viewpoints; hence saliency of the 2D features is crucial to maintain a good tracking quality during an AR session. To control the quality of the features, the parameters of the feature detector: the Shi Tomasi score (Shi and Tomasi, 1994) and non-maximum suppression threshold were carefully and empirically tuned onsite to yield the best tracking results.

For this application, a variant of PTAM named PTAMM (Castle et al., 2008) was chosen because of its ability to use multiple maps and switch between maps without the need for re-initialization. The ability of bypassing the initialization step at startup proved useful as it allows for scale and anchoring preservation.

4.3.2. *Scale estimation*

A fundamental shortcoming underlying monocular based localization and mapping solutions is the fact that the obtained maps and motion are solved up to an unknown scale. To perceive relative depth, monocular Visual SLAM systems rely on temporal stereoscopy of the same camera as it observes the scene from different viewpoints. This requirement enforces the need for a manual initialization step to recover the poses of the first two keyframes. Since the absolute motion between the first two keyframes cannot be recovered, the relative motion magnitude recovered in the initialization procedure is used to scale the map; this however means that different initializations of the system lead to different scales. Fortunately, PTAMM proved useful in this regard as it allowed for bypassing the initialization procedure when a previously generated map is loaded in the system. The scale loss problem is then reduced to estimating the scale during the scene exploration phase only and saving it for later use when the map is later invoked.

We chose to recover the absolute motion between the initializing keyframes by introducing a temporary artificial marker in the scene of known dimensions. The ARUCO library (Garrido-Jurado et al., 2014) was used to generate a binary marker that allows for the camera pose to be recovered in metric scale whenever the marker is visible in the frame.

4.3.3. *Anchoring*

A vital part in an augmented reality application is the seamless integration between virtual objects and the actual scene. Proper anchoring (alignment) of the objects is mandatory to achieving a minimum level of expected realism. Fig. 7 shows the situation where neither the scale is estimated nor proper anchoring is established. The result is a misplaced virtual object with wrong dimensions. To make matters worse, PTAMM's coordinate frame in chosen at random (at the mean of the initial scene) on startup and therefore cannot be manually pinned to a given location.

To properly anchor the virtual objects in the actual site, we make use once more of the introduced marker. As shown in Fig. 8, during regular system operation, for any given frame that observes the board of markers, both T_I^P and T_I^M are available allowing the recovery of the transformation T_M^P as long as both coordinate frames have the same

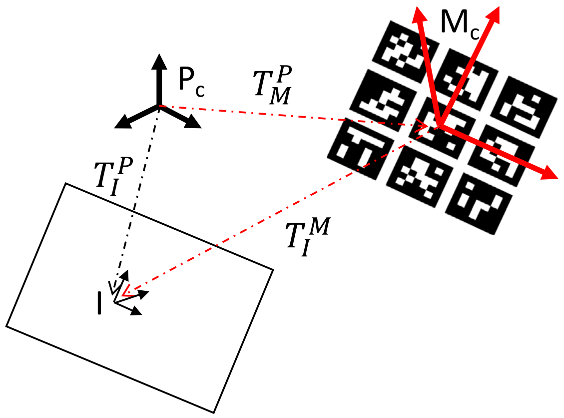


Fig. 8. The anchoring transform relating the marker's space to PTAMM's space.

units (*i.e.,* PTAMM's map is scaled).

The transformation T_M^P is of great importance as it allows us to define the pose for any point in the scene where the marker is placed, with respect to PTAMM's coordinate frame. Fig. 9 summarizes the anchoring process once T_M^P is found. The origin of the virtual objects is chosen such that it coincides with its corresponding location in the real site at which the marker board is placed hence ensuring proper alignment between the objects.

4.3.4. *Lighting conditions*

Despite the significant amount of research in the computer vision community, lighting conditions are still considered a major challenge to most vision-based systems. Both lighting magnitude and direction affect the conversion in a non-trivial way and can greatly degrade the performance of a Visual SLAM system. While the problem can be addressed in indoor environment by controlling the light source, outdoors environments are plagued with lighting factors that cannot be controlled.

In an effort to gain immunity against light magnitude changes, PTAMM removes the average intensity of every frame from the individual pixel values. Unfortunately, such a strategy is futile when the direction of the lighting source varies, (*i.e.*, when the sun's position in the sky changes). To address this setback, the relative small size of the scene is exploited to generate a database of maps by repeating the scene exploration phase at different times of the day, each covering different lighting conditions (sunny, cloudy) and different sun positions in the sky. The collected data is saved into separate maps and re-loaded during an AR session at startup. Failure recovery is then initiated to query from the database of maps a keyframe that is closest (in terms of appearance) to the currently observed frame. The system then proceeds by tracking the camera's pose within the corresponding map. Whenever the light source changes significantly, tracking is lost and the above-mentioned procedure is repeated to invoke another map that is best suited for the new lighting conditions.

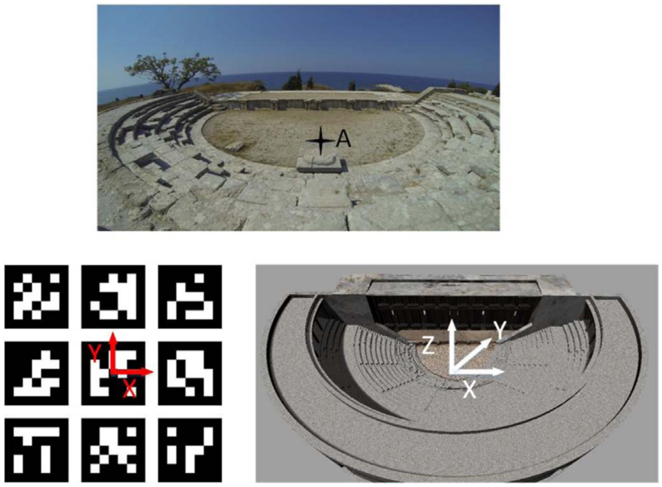


Fig. 9. The board's center is placed at point A in the real site and its location is recorded within PTAMM's map; this location is set as the origin of the virtual model.

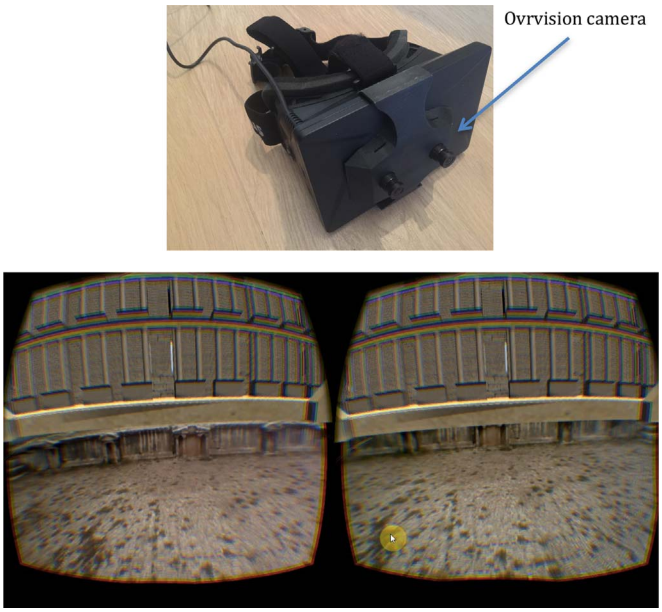


Fig. 10. (top) Hardware used in the AR experience: The oculus headset with Ovrvision stereo rig attached. (bottom) The hypothesized model augmented on the video feed of the stereo-cameras as seen by the user through the oculus displays.

4.4. *Visualization of the AR*

Once the user's pose can be tracked within the scene, the final step for completing the AR system is the projection of virtual objects onto the user's view. The augmentation may take place on a 2D screen or a head mounted stereo display.

During on-site experiment we noted that in bright sunlit conditions, the screens on see-through glasses such as the Moverio[6] were not visible. The only alternative was to use completely immersive headsets such as the Oculus RIFT,[7] and stream the live video through a camera (Ovrvision[8]) mounted on the face of the RIFT (See Fig. 10).

5. Interacting with the augmented model in a virtual setting

5.1. *Creating the virtual world in unity*

The virtual reality world was developed using the Unity engine,[9] which is a cross-platform game engine used to develop video games for PC, consoles, mobile devices and websites. The developed application allows users to visualize and interact in a walkthrough of a 3D real site. Furthermore, to make the application more interactive, a guide was added to the scene designed to attract the users' attention and give them a virtual tour of the site.

The development of the 2-D virtual application first started by adding a player prefab into a unity scene; a unity scene is a virtual space that includes all "game" objects whereas a player prefab is a game object that keeps track of all the information regarding the "player"; it defines what to be rendered on the player's screen by keeping track of

---

[6] www.epson.com/moverio.

[7] https://www.oculus.com.

[8] ovrvision.com.

[9] https://unity3d.com/.

Fig. 11. The hypothesized model added and aligned on top the computerized model of the theater.

its virtual pose. The virtual pose of a player is updated through user's input on a keyboard (to change the position) and a mouse (to change the orientation) or alternatively a gaming controller (an x-box controller).

Once the player was setup in the scene, we ported the generated textured mesh of the actual site into the scene. To allow for user interaction with the mesh, we manually added physical colliders that interact with the users' virtual pose. The hypothesized model (Fig. 11) was then added into the scene and carefully merged with the actual site.

5.2. Visualization

The entire virtual reality application was viewed on an Oculus Rift. The Oculus contains an IMU (inertial measurement unit) made of an accelerometer, a gyroscope and a magnetometer. The purpose of the IMU is to precisely track the users' head pose as they rotate around and accordingly adjust the player's prefab information. By such integration, the 3-D application responds to the actual motion of the users' head thus replacing the need for a mouse or orientation controls as they are provided by the IMU. However the Oculus' IMU cannot provide position information and hence the need for a keyboard or a gaming controller remains. The oculus connects to a pc or a laptop through an HDMI connection and a USB cable.

The interaction with the augmented cultural heritage model while the user is off-site holds tremendous power in drawing users and potential tourists to explore the site and plan their visits and tours accordingly. It is therefore important to develop site models that are visually realistic, allow a user to navigate through the site to explore its various features, and can engage the user by adding information to an otherwise passive experience.

5.3. Environment modeling

The starting point for developing a realistic stereoscopic model suitable for an immersive experience is to augment the site reconstruction with an outdoor environment model including sky, sea, and appropriate sunlight conditions. Since the user is not expected to have any meaningful interaction with the sky or the sea surrounding the Byblos Theater, simple generic models for them were suitable. Lighting however presented a more consequential challenge. Without an appropriate lighting and shading model, elements of the virtual site may present conflicting cues to the user since different portions of the 3D model reconstruction was generated from photographs taken under differing lighting conditions. The timings of the image capture were chosen to allow, for every image, fully lit views with minimal darkened areas due to shading. For the different spatial locations of the site, this occurs at different parts of the day when the sun is different locations. In the reconstruction, this creates lighting artifacts that have to be masked in the final display to avoid inconsistently lit scenery. For this, special light sources were manually designed and introduced in the scene and corresponding lightmaps were generated. Lightmaps allow the expensive rendering and lighting computations to be done in texture space offline and enable real time display of the properly illuminated triangle mesh (Fig. 12). The enclosed video shows the quality of the reconstructed Byblos augmented with environment models and enhanced lighting.

Fig. 12. Lighting and shading effects rendered on the 3D model.

5.4. Obstacle and hazard avoidance

The immersive 3D augmented stereo-displayed world has one severe limitation when it comes to navigating it. The straightforward way to navigate in this world is to equip the user with a pad so he/she can turn around and move forward or backward by pushing on a joystick. However it turns out that stepping over a ledge or other surface in the virtual world gives a strong illusion of actually falling down in the real world. This has been documented in the VR and gaming literature. The perception is in fact so strong that people actually fall to the ground if they are standing up or become extremely uncomfortable even if sitting down, ruining the experience of navigating the site.

The environment of Byblos has a stage and many stairs and steps that represent navigation hazards that users can “fall” from. It was therefore essential to design an obstacle avoidance system to prevent the user from crossing over surfaces with a significant height difference. The standard way to build an obstacle avoidance system is to build a navigation mesh, which consists of the collection of polygons approximating the walkable surfaces of the site. Constraints on height differences of neighboring polygons the user can step between, steepness of slopes they can go down or up, and similar local geometric thresholds are built into the navigation mesh defining safe traversal paths. In the case of the Byblos site model, this approach was not practical because the reconstruction has a relatively high polygon count representing the actual bumpy surface of the current site. In order to obtain a scalable solution, we generated a low polygon count version of the theater and overlaid it on the model. The navigation mesh is defined on this low polygon count model to make the navigation

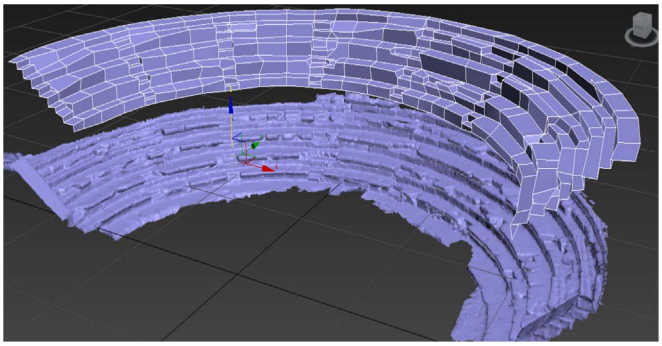


Fig. 13. Reconstructed model and low-polygon approximation used for navigation and hazard avoidance. The low-polygon approximation is invisible during interaction.

Fig. 14. The virtual guide along with the path rendered in the scene to help guide the user.

computations feasible. The low polygon model is only used for computations and is not rendered on the screen. Fig. 13 shows the reconstruction and the coarse approximation.

5.5. Guidance and navigation

To make the application more interactive, a virtual guide (Fig. 14) was added to the scene. Specific POIs were selected near historical objects of interest, such as the theater's altar. The guide floats around the predefined checkpoints and waits for the user to get within a small distance from it before explaining what the object of interest is. Once the user is within the vicinity of the object, it glows, capturing the user's attention before the guide starts narrating informative audio about the object. Once finished, the guide moves to the next checkpoint and waits for the user to get near again. To help show the user where to go, we employed an A* path planning algorithm that determines a path between the user and the guide and renders it on screen for the user to follow. Note that the user is not obliged to follow the guide and is free to roam around the scene.

To create the storyboard, the History Department at the American University of Beirut was consulted in order to determine the Points of Interest (POI) inside the theater. According to the safety guidelines of the Oculus and to reduce the risk of uncomfortable feelings inside the virtual world, the complete guided tour was designed to last five minutes. Accordingly, ten points of interests were selected inside the theater, with a description of approximately thirty words per POI.

6. Conclusion

This paper presented a regionally-guided hypothesis about a possible reconstruction of the Byblos Roman theater; different than previous works (El-Khoury et al., 2006), the reconstructed model was used in the design of an end-user virtual and augmented reality applications for touristic purposes in the archeological site.

The technological challenges and limitations of both virtual and augmented methods employed were detailed and systematically addressed yielding a scalable approach that can be extended to other sites.

We are currently testing our application in Byblos and soliciting feedback from users with respect to their experience. Accordingly, we hope to address any notable issues in the second generation of our applications. We are also currently planning the development of similar applications in other archeological sites in Lebanon.

Acknowledgment

This work was funded by the IAM grant number I-A/1.2/113 of the European Neighborhood Partnership Instrument (ENPI). We would like to acknowledge Dr. Paul Newson and Dr. Nadia Maria El-Cheikh from the Department of History and Archaeology at the American University of Beirut for preparing the narrative for the virtual tour guide. We would also like to acknowledge the Directorate General of Antiquities (DGA) of Lebanon for their assistance in preparing the narrative.

Appendix A. Supplementary material

Supplementary data associated with this article can be found in the online version at http://dx.doi.org/10.1016/j.daach.2017.03.002.

References

Castle, R.O., Klein, G., Murray, D.W., 2008. Video-rate localization in multiple maps for wearable augmented reality. In: Proceedings of the 12th {IEEE} Int Symp on Wearable Computers. Pittsburgh PA, Sept 28–Oct 1, 2008. pp. 15–2.

Dunand, M., 1954. Fouilles de Byblos: 1933–1938., Études et Documents d'Archéologie. Librairie d'Amerique et d'Orient, Adrien Maisonneuve.

Durrant-Whyte, H., Bailey, T., 2006. Simultaneous Localisation and Mapping (SLAM): Part I the essential algorithms. IEEE Robot. Autom. Mag. 2, 2006.

El-Khoury, N., Giovanni, D., Dorta, T., 2006. Digital reconstruction as a means of understanding a building's history - case studies of a multilayer prototype. In: Communicating Space(s): 24th eCAADe Conference Proceedings, eCAADe: Conferences. University of Thessaly, Volos, Greece.

Garrido-Jurado, S., Muñoz-Salinas, R., Madrid-Cuevas, F.J., MarÃn-JimÃ©nez, M.J., 2014. Automatic generation and detection of highly reliable fiducial markers under occlusion. Pattern Recognit. 47, 2280–2292. http://dx.doi.org/10.1016/j.patcog.2014.01.005.

Jidejian, N., 1971. Byblos Through the Ages. Dar el-Machreq Publishers, Distribution Librairie Orientale.

Scaramuzza, D., Fraundorfer, F., 2011. Visual odometry [Tutorial]. IEEE Robot. Autom. Mag. 18, 80–92. http://dx.doi.org/10.1109/MRA.2011.943233.

Shi, J., Tomasi, C., 1994. Good features to track. In: Proceedings of the 1994 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR'94. pp. 593–600. ⟨http://doi.org/10.1109/CVPR.1994.323794⟩.

Younes, G., Asmar, D., Shammas, E., 2016. A Survey on non-filter-based monocular visual SLAM systems. ArXiv e-prints/1607., 00470.