Integrating Terrain Data into Virtual Reality Systems for Outer Space Exploration

Giuseppe Lorenzo Catalano[1(✉)], Eugenio Topa[2], and Agata Marta Soccini[1]

[1] University of Torino, via Verdi 8, Torino, Italy
giuseppe.catalano375@edu.unito.it, agatamarta.soccini@unito.it
[2] ALTEC S.p.A., Corso Marche 79, Torino, Italy
eugenio.topa@altecspace.it

Abstract. Virtual reality applications are extensively used in the space exploration domain to support different tasks that vary from astronaut training to mission planning, to system design and engineering. A key role is played by the interactive visualization of scientifically-accurate data related to outer space elements, such as, among others, stars, galaxies and planets. In the current work, we propose a method for the 3D visualization of data related to terrain surfaces of different celestial bodies, and the integration of the terrain models into a space-exploration virtual reality application. In particular, we provide insights into the pipeline we used to import and process the terrain data, coming in the form of Digital Elevation Models, and the techniques to visualize and geo-reference them inside the application. The proposed methodology showed accuracy in the visual outcome of the reconstructed surfaces, as well as positive results in terms of software performance of the real-time application.

1 Introduction

Virtual Reality (VR) applications have been widespread for years in the space industry, which took great advantages by adopting these solutions; in many cases they have become the *de facto* standard, as they are able to bring a great reduction in costs, manpower, and required skill levels [1]. Numerous tasks are being successfully carried out using VR applications. It is worth mentioning, among others: I) astronaut training and operational assessments [2]; II) remote operations using safe, virtual environments that help in enhancing the operators' responsiveness and focus [3]; III) system design and manufacturing tools that relieve companies of investing large amounts of economic and human resources [4]; IV) diagnostics and analysis using collaborative frameworks that interface in real-time with the monitored systems [5,6]. Moreover, VR applications can also be used for scientific research and outreach [7]. With this regard, virtual simulations of the universe are built to explore huge amounts of data that are being collected by spacecraft, such as the star catalogues provided by the European

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2024
L. T. De Paolis et al. (Eds.): XR Salento 2024, LNCS 15027, pp. 221–229, 2024.
https://doi.org/10.1007/978-3-031-71707-9_16

Space Agency (ESA)'s Gaia mission [8] or the archive of exoplanets provided by the National Aeronautics and Space Administration (NASA) [9]. Scientific data can also be used to face the need of precise information for accurate and realistic simulations when planning new missions [5]. The goal of the current work is to present a methodology to introduce new kinds of data into VR space simulations. In particular we focused on the representation of terrain surfaces of celestial bodies, given in the form of Digital Elevation Models (DEM), with the objective of visualizing terrains as similarly to their real counterparts as possible, without compromising the performance of the system. The possibility to recreate surfaces of celestial bodies would indeed allow the system to offer better mission planning capabilities, as well as simulations of rovers and other spacecraft. In this scenario we defined our research question as follows: How can we integrate new data related to celestial body terrains, into a VR system primarily designed for visualizing star catalogues?

2 Related Work

Several Virtual Reality tools for the visualization of scientific data have been developed. These tools rely on a series of techniques for exploring the simulation in real-time while processing large amounts of data. A notable example is Gaia Sky [10], an open-source tool used for visualizing and analyzing the entirety of the Gaia star catalogue. Another virtuous example is OpenSpace [11], an open-source framework for representing in details planets, stars and galaxies, on-going missions and more. OpenSpace aims to provide a versatile system that can be efficiently used in astronomy research, mission planning, and data communication. Other similar examples exist, however they with provide smaller amounts of data resulting in an overall limited usage for scientific purposes. Some example are Celestia[1] and Space Engine[2]. Elevation models also have their importance for a number of tasks. Detailed representations of extraterrestrial surfaces can be used for mission planning and simulation endeavours [12]. Moreover, their capabilities of accurately representing real surfaces have been used in frameworks that can help in testing autonomous navigation algorithms [13], and they have been even employed for training deep learning models which, given the lack of real terrain images, rely on pictures taken from virtual environments [14]. In this scenario, Astra Data Navigator, the application discussed in the current paper, is a system focused primarily on the scientific side of the simulation. The software gives support for the visualization of different star catalogues and the interaction with the datasets directly inside the scene.

3 Astra Data Navigator System

Astra Data Navigator (ADN) is a VR application developed by ALTEC S.p.A.[3], a company active in a number of space exploration missions. ADN allows for real-

[1] https://celestiaproject.space/.
[2] https://spaceengine.org/.
[3] https://www.altecspace.it/.

time visualization and navigation of star catalogues and other celestial objects. ADN was developed using Unity3D[4] LTS 2022.3.7f1, and provides several features.

1. The visualization and representation of data coming from different star catalogues such as Hipparcos [15], Tycho-2 [16] and Gaia DR2 [17]. Indexed structures for efficient data retrieval, along with a custom shader for lightweight rendering of the stars, are used; the rendering is performed according to the scientific parameters of the stars, such as radius and surface temperature.
2. A time simulation that calculates rotation around the axis and revolution around the Sun for the celestial objects in the Solar system, such as planets, satellites and spacecraft. The position of an entity at a specific time can be retrieved in ADN via the use of SPICE kernels[5], calculation tools developed by NASA to give extremely precise information for several celestial objects.

ADN allows the free and seamless navigation of the universe by using the 'floating origin' approach [18], which consists in moving the entire scene while keeping the camera fixed at the origin and thus avoiding approximation errors over large scales. ADN also offers different visualization modes and tools manipulating the representation of the loaded dataset; these features, along with the wide compatibility given by the use of the Unity engine, make ADN suitable for being an effective framework that can be used for scientific data visualization, and that can be further expanded for dealing with more kinds of data. ADN, at the current state, runs on desktop computers (both Windows and MacOS operating systems were tested), with additional support for stereoscopic displays and a port for Head-Mounted Displays in development.

4 Digital Elevation Models in Astra Data Navigator

This section describes the work done to integrate elevation data inside Astra Data Navigator. Figure 2 provides a visual representation of the application's general workflow, describing a pipeline for importing and visualizing the data on the celestial bodies' surfaces.

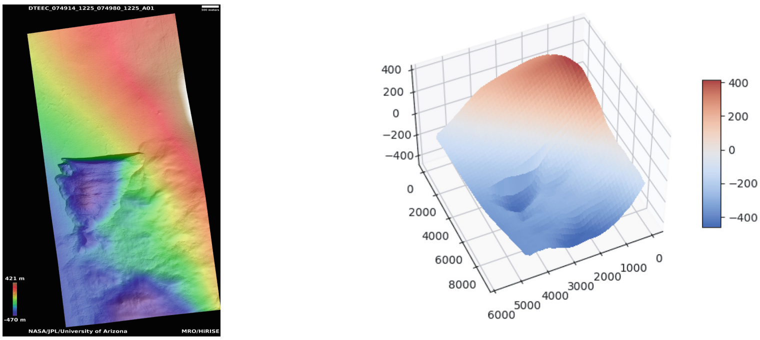
4.1 Simulation Re-scaling

Before the inclusion of elevation data inside ADN, a critical part of the development consisted in re-scaling the entire simulation to the meters. Indeed, previously ADN had been scaled in a way such that every unit of the virtual environment would correspond to 1 km inside the simulation; this would however not fit the purposes of the new functionalities, whose main goal was to represent data accurate to single meters. The simulation underwent therefore a work of re-scaling, which concerned all the parameters related to positions and velocities, aimed at bringing each unit of the virtual environment to correspond to 1 m inside the simulation.

[4] https://unity.com/.
[5] https://naif.jpl.nasa.gov/naif/.

4.2 Retrieval and Preprocessing of DEM Raw Files

Digital Elevation Models (DEMs) are digital raster images representing surface terrains of a particular celestial body, made up of pixels which typically store signed 16-bits integer or 32-bit floating-point values; each pixel contains a value that represents an altitude. DEMs, which can also be thought of as maps, come accompanied by a series of metadata, such as coordinates (latitude, longitude), coordinate system specifications and map resolution (typically expressed in meters per pixel), that allow to uniquely identify the portion of the celestial body covered by the surface data provided by the DEM. DEMs may range in size from a handful of Megabytes to even Gigabytes, depending on their resolution and the area they cover: they can be either global, where the model covers the entirety of the body's surface and usually have a resolution in the order of hundreds of meters per pixel, or local, where only a specific area is covered in more details, with a resolution that can arrive to less than a meter per pixel; Fig. 1 shows a local DEM from Mars.



(a) (b)

Fig. 1. Local DEM from Mars, viewed as a colormap (a) and as a 3D surface (b).

Numerous DEM data sources exist, mostly covering Earth but also Mars, thanks to the HiRISE orbiter camera [19], and the Moon, thanks to the LROC orbiter camera [20]. DEMs may come in different file formats and metadata specifications; the first part of the pipeline consisted therefore in creating a preprocessing phase, that could bring all the different files to a common interface that would be recognizable by ADN. Using Python (3.10) we developed a library to operate on datasets of DEM files. The preprocessing converts all the files to GeoTIFF, a common format for maps, and read by ADN using the LibTIFF library[6], as well as extrapolating the most useful metadata and storing it into JSON files. It is noteworthy to point out how, during preprocessing, the original raster data is left untouched, and therefore retains all the raw information. The

[6] https://bitmiracle.com/libtiff/.

scripts heavily rely on GDAL[7], a library that allows to operate with ease on geospatial datasets and maps and is able to perform a series of coordinates transformations, as well as converting files from one format to another. During development, GDAL 3.7.1 was used.

4.3 DEM Geo-Referencing

When ADN is started, DEMs metadata are loaded so that the simulation becomes aware of their presence and starts keeping track of their position. A critical aspect resides in geo-referencing the DEM on a celestial body's surface; that is, identifying the correct area covered by the DEM on the surface. This is needed because, instead of creating a static model of the DEM beforehand that is loaded separately, the solution adopted consists in importing the raw DEM data directly into ADN, using that data to dynamically deform the celestial body's mesh at runtime. Celestial bodies are mostly treated as spheres in ADN; this approximation, despite producing some representation errors, allows to treat latitude and longitude coordinates as Euler angles that, along with the map's size and resolution, correctly identify the region of a specific local DEM. This procedure is enough to identify the correct region for most of the DEMs, projected to the image plane such that the north is always headed towards the top of the raster; particular attention needs however to be paid for those areas close to the poles, whose DEMs are obtained using a different map projection. The geo-referencing phase takes into account such cases by applying a further rotation, equal to the longitude value, so that the north is always pointing in the right direction.

4.4 Landing Mode and Dynamic Memory Management

The new version of ADN allows the user to 'land' onto celestial bodies and remain stationary on the surface. Upon entering landing mode, the data of the global DEM for that particular entity is loaded into memory; moreover, as the user is navigating the surface of a celestial body, ADN constantly monitors the camera position with respect to the available DEMs. When approaching a particular local DEM its data is fully loaded into memory, so that it can be always available for being sampled; when leaving the data is deallocated, preventing the system memory from being saturated by entire chunks of DEM data. Memory deallocation does not affect the monitoring of the DEM's status by ADN and, if the conditions are met again, will try to load the DEM data again.

5 Evaluation and Results

The new functionalities added in ADN have introduced the possibility to handle a new type of data. A qualitative overview of the reconstruction results is

[7] https://gdal.org/.

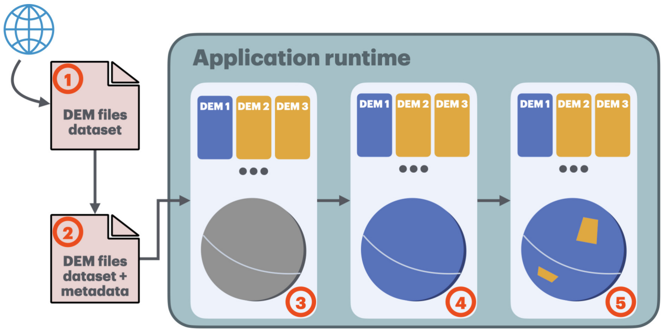


Fig. 2. Pipeline for visualizing elevation data inside Astra Data Navigator. 1) DEM files are downloaded from online repositories and stored locally. 2) DEMs are preprocessed for metadata extraction. 3) DEMs metadata are loaded in the application runtime (blue is global DEM, yellow are local DEMs). 4) When landing on a celestial body, the global DEM data is loaded. 5) When getting close to the location of specific local DEMs, their data is also loaded. (Color figure online)

provided by Fig. 3, by comparing pictures portraying regional sites of interest and their respective result inside ADN. The images show that ADN is capable of recreating the surfaces up to the highest details available via the DEM

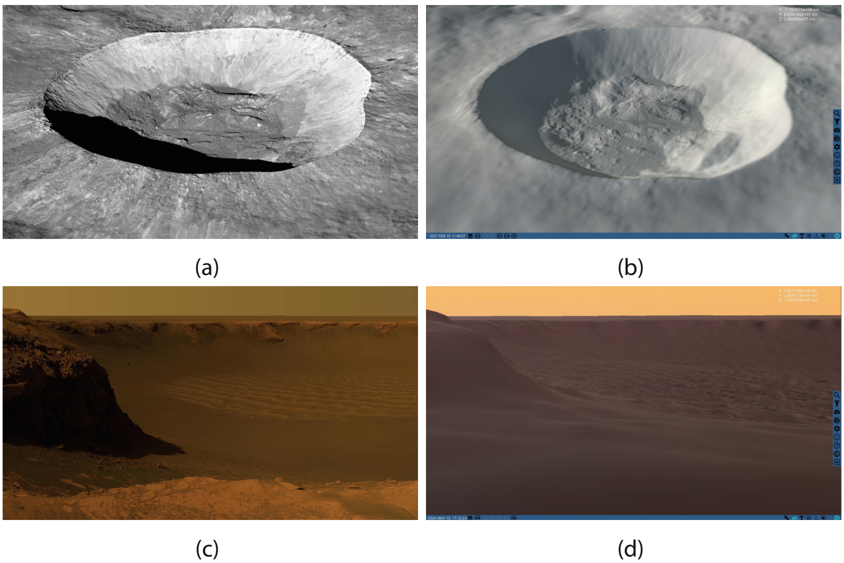


Fig. 3. Comparisons between real images (left) and the corresponding reconstructions in ADN (right). Figures 3a and 3b are from the Giordano Bruno crater on the Moon, whereas Figs. 3c and 3d are from the Victoria crater on Mars.

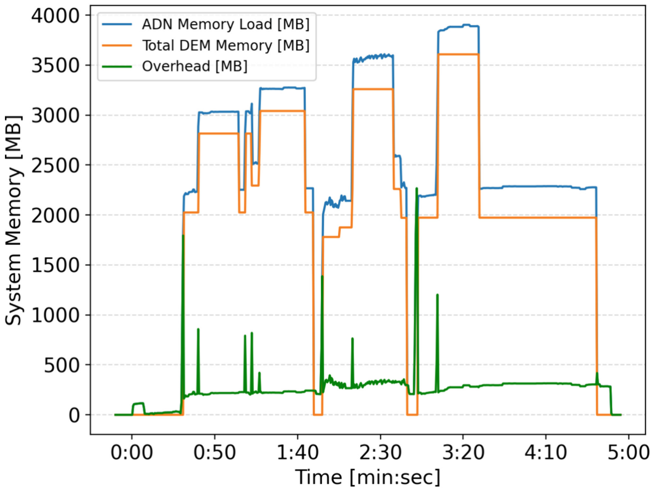


Fig. 4. System memory usage during a test run of the new version of ADN.

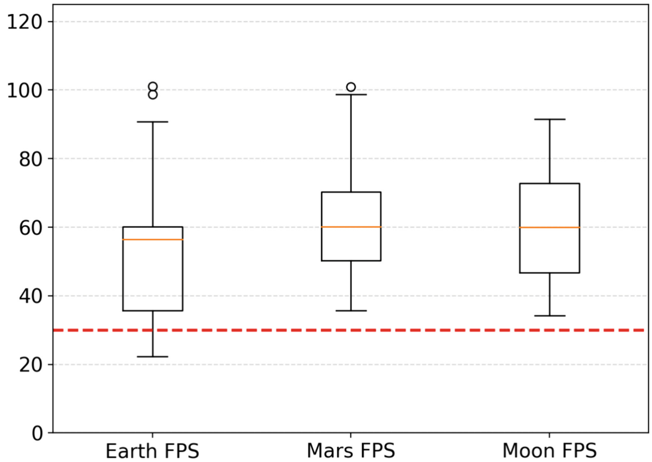


Fig. 5. FPS distribution during a test run of the new version of ADN.

data, demonstrating the effectiveness of the preprocessing stage (Sect. 4.2) and the accuracy in geo-referencing and mesh deforming (Sect. 4.3). With regard to performance analysis, one first critical aspect we evaluated is the usage of system memory (as described in Sect. 4.4): DEM datasets can grow very rapidly in size, therefore it was necessary to verify whether the implementation would prevent the system from being saturated. Figure 4 shows that, over different the total overhead remains overall constant at around 300 MB, suggesting that the implementation could scale depending on the dataset size. Another performance metric we evaluated is the distribution of frames-per-second (FPS) obtained during a test run that visits all the celestial bodies with DEMs (Earth, Moon and Mars); the plots in Fig. 5 show good results for Mars and Moon, with a median value at 60 FPS and never dipping below 30 FPS, whereas some hiccups occur while visiting Earth (which is the biggest one of the celestial bodies taken into exam). We conducted the benchmark on a system with 2x Intel Xeon E5-2650 CPUs, an NVIDIA Quadro K5200 GPU, and a total RAM memory of 32 GB (2133 MHz).

6 Conclusions and Future Work

In this paper we initially presented Astra Data Navigator, a VR application for the navigation of the universe, that includes large amounts of scientific data coming from star catalogues. We then presented our main contribution, that is a methodology we developed to process DEM data and integrate them into the simulation. Imported DEMs are then used to reconstruct the surfaces seamlessly, directly onto the surfaces of the celestial bodies. On top of that, we designed and developed new functionalities, capable of handling and representing s new type of data, related to terrain surfaces of different celestial planets. This work shows how this kind of system can be used as a common framework for developing an integrated environment for the scientific community, providing a platform with tools suitable for mission planning, rover testing, and studies of planetary terrain conditions. Further developments are currently under development, namely for simulating the behaviour of spacecraft such as orbiters and rovers, therefore expanding even more the variety of tools available.

Acknowledgements. Supported by the National Operative Program (PON) on "Research and Innovation" 2014–2020 of the Italian Ministry of University and Research, and by HST Center (Human Sciences and Technologies) at University of Torino, and by ALTEC S.p.A.

References

1. Pirker, J.: The potential of virtual reality for aerospace applications. In: 2022 IEEE Aerospace Conference (AERO), pp. 1–8. IEEE (2022)
2. Dufresne, F., et al.: Touching the moon: leveraging passive haptics, embodiment and presence for operational assessments in virtual reality. In: Proceedings of the CHI Conference on Human Factors in Computing Systems (2024)
3. Laskey, L., et al.: Evaluating the effectiveness of game-based virtual reality in satellite ground control operations education and training. Int. J. Aviat. Aeronaut. Aerosp. **11**(1), 1 (2024)
4. Phanden, R.K., Sharma, P., Dubey, A.: A review on simulation in digital twin for aerospace, manufacturing and robotics. Mater. Today Proc. **38**, 174–178 (2021)
5. Soccini, A.M., et al.: 'IXV-trajectory' and 'IXV-asset': virtual reality applications for the aerothermodynamics analysis of IXV. In: IEEE Virtual Reality (VR), pp. 397–398. IEEE (2015)
6. Soccini, A.M., et al.: Virtual reality interface for multidisciplinary physical analysis of space vehicles. In: EuroVr (2014)

7. Ynnerman, A., Löwgren, J., Tibell, L.: Exploranation: a new science communication paradigm. IEEE Comput. Graph. Appl. **38**(3), 13–20 (2018)
8. Prusti, T., et al.: The Gaia mission. Astronomy Astrophys. **595**, A1 (2016)
9. Akeson, R.L., et al.: The NASA exoplanet archive: data and tools for exoplanet research. Publ. Astron. Soc. Pac. **125**(930), 989 (2013)
10. Sagristà, A., et al.: Gaia sky: navigating the Gaia catalog. IEEE Trans. Vis. Comput. Graph. **25**(1), 1070–1079 (2018)
11. Bock, A., et al.: OpenSpace: an open-source astrovisualization framework. J. Open Source Softw. **2**(15) (2017)
12. Piovano, L., et al.: Virtual reality representation of Martian soil for space exploration. Pattern Recogn. Image Anal. **23**, 111–129 (2013)
13. Villa, J., Mcmahon, J., Nesnas, I.: Image rendering and terrain generation of planetary surfaces using source-available tools. In: Proceedings of the 46th Annual AAS Guidance, Navigation & Control Conference, Breckenridge, CO, USA, pp. 1–24 (2023)
14. Wu, B., et al.: Absolute localization through orbital maps and surface perspective imagery: a synthetic lunar dataset and neural network approach. In: 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 3262–3267. IEEE (2019)
15. Perryman, M.A.C., et al.: The HIPPARCOS catalogue. Astronomy Astrophys. **323**, L49–L52 (1997)
16. Hog, E., et al.: The Tycho-2 catalogue of the 2.5 million brightest stars. Astronomy Astrophys. **363**(1), 385–390 (2000)
17. Brown, A.G.A., et al.: Gaia data release 2-summary of the contents and survey properties. Astronomy Astrophys. **616**, A1 (2018)
18. Thome, C.: Using a floating origin to improve fidelity and performance of large, distributed virtual worlds. In: 2005 International Conference on Cyberworlds (CW 2005), pp. 8–270. IEEE (2005)
19. Mcewen, A.S., et al.: Mars reconnaissance orbiter's high resolution imaging science experiment (HiRISE). J. Geophys. Res. Planets (2007)
20. Robinson, M.S., et al.: Lunar reconnaissance orbiter camera (LROC) instrument overview. Space Sci. Rev. **150**, 81–124 (2010)