Visualization of climate simulation data in virtual reality using commercial game engines

Daniel Kolb[1], Wolfgang Kurtz [1], Jens Weismüller[1], Alexander von Ramm [1], Ralf Ludwig [2], Dieter Kranzlmüller[1]

1. Introduction

Due to the size of its customer base, the video game industry has long been a well-funded proponent of innovative real-time computer graphics. Many advancements in the field of computer graphics, software and hardware, have become cost-effective due to their use in video games, which in turn funded even further research and breakthroughs. Recent changes in the monetization of commercial game engines [1] made their use in less revenue driven institutions affordable and, hence, possible. This allows us, given suitable hardware, to build and run computationally expensive fully interactive real-time visualizations at a fraction of the cost and time that was required previously. We can thus investigate and explore the data in our application far sooner. Additionally, we are able to spend significantly more time on iteratively refining the user interaction as well as the preprocessing of the raw scientific data.

Scientific visualizations using virtual reality have previously proven to be a potent tool in the presentation and transfer of knowledge; be it geo-physics [2], archeology [3], genomic analysis [4] or engineering [5]. We aim to present a further proof of concept regarding the use of virtual reality in climate visualizations [6] by visualizing data provided by the ClimEx project [7].

The ClimEx project is an international collaboration between research facilities, universities and public water agencies in Bavaria and Quebec. It investigates the effects of climate change on meteorological and hydrological extreme events and assesses implications for water management in the two regions. Within this project, an ensemble of 50 transient runs of the regional climate model CRCM5 have been run at approximately 11km resolution for two domains in Europe and North America, resulting in 7500 years of modelled climate for each domain. As each of these runs is initialized with only slightly altered starting conditions, the ensemble can be interpreted as modelled natural variability. From a reanalysis run of the model, we extracted the Pentecost flood in Southern Germany and Austria in May 1999. From the European ensemble, we selected two heavy precipitation events from different members in the 2060s and the 2080s. In this work, we visualize these three rainfall events over a 3D representation of Bavaria.

Our goal is to provide a detailed description of the steps necessary to create an immersive 3D visualization of climate data. We also describe an interaction model which empowers the user with many ways to manipulate the virtual scene, while retaining a simple and intuitive usability.

[1] Leibniz-Rechenzentrum, Munich Germany, corresponding author‘s contact. daniel.kolb@lrz.de
[2] Ludwig-Maximilians-Universität, Munich Germany

2. Pre-processing of spatial input data for the game engine

The required components of the visualisation consisted of (1) a digital elevation model (DEM) that forms the baseline of the visualised domain, (2) high-resolution satellite images that provide the texture for the DEM, and (3) the accumulated precipitation data from the ClimEx simulations for three events. In the following, we describe the pre-processing steps of the spatial input data (DEM, satellite images and simulation results). The pre-processing of the digital elevation model (DEM) and the simulation data was realized with the statistical software package R[3]. We used the software package GDAL[4] for the preparation of the satellite image for the texture, Autodesk 3ds Max[5] for the composition of the DEM and its texture and Blender[6] for refining the simulation data.

2.1 Digital elevation model

We derived the digital elevation model for the visualisation domain (Bavaria) from the EU-DEM v1.1 product provided by the European Environmental Agency [8]. This data set is freely available and provides a Europe-wide DEM with a spatial resolution of 25 m in the reference coordinate system EPSG 3035. We downloaded two tiles (100x100km) of this data set (partly covering Bavaria) from the EEA website and merged them together using the functionality provided by the R-package rgdal [9]. We then cropped the merged DEM with the contour of Bavaria, which is available from the State Surveying Office of Bavaria [10]. As the next step, we transformed the so derived grid-based DEM information to an object-based format that can be handled by the game engine. We accomplished this by subdividing each grid cell into two triangles. The vertices of each triangle are defined by the corresponding corners of the respective grid cell. This information was then stored in the Wavefront obj file format. Since the large file size of the single DEM obj file caused severe problems for the further processing within the game engine, we split the visualisation domain orthogonally into equally-sized subdomains. Each of these subdomains produced a separate Wavefront obj file.

We imported the thus generated 3D files to 3ds Max, where we used the provided ProOptimizer function to reduce the vertex count of the 3D model to 6.25%. However, we excluded bordering vertices from this operation, so that the resulting simplified subdomains continued to fit together. In preparation for the fitting of the texture, we also generated a single planar UVW-Map for all subdomains.

2.2 Satellite images as texture

The texture of the visualisation was realised with freely available satellite images from the SENTINEL-2 platform. We downloaded the visible colour bands (red, green, blue) from tiles covering the visualisation domain from the SENTINEL access hub Remote Pixel. For each of the tiles, we selected the most suitable

[3] https://www.R-project.org/.

[4] http://gdal.org

[5] https://www.autodesk.eu/products/3ds-max/overview

[6] https://www.blender.org/

satellite image by manually scanning the time series of available data for a minimum cloud coverage within the spring months of 2017 and 2018. We then processed the downloaded tiles with the software package GDAL. As a first step, we merged the three visual bands separately for each tile. In a second step, we merged the tiles with the same projection (different UTM zones) together, projected the so derived composites (for each UTM zone) to the same UTM zone and merged again. Finally, we projected the image to the same reference coordinate system of the DEM (EPSG 3035, see above), cropped to the extent of the DEM and converted to an 8-bit portable network graphics (PNG) file.

Afterwards, we added the completed texture to the previously prepared group of subdomains in 3ds Max. We then used the Render to Texture function to create individual texture snippets for each subdomain. Each snippet was baked with the maximum possible dimensions of 8192 * 8192 pixel to retain the fidelity of the original texture. Thereafter, we assigned each texture snippet to its DEM subdomain and exported the combined results to individual obj files.

2.3 Simulation data

Data on simulated precipitation from the ClimEx project are available in netCDF format and were read into R with the ncdf4 package[7]. We first projected the data to the reference coordinate system of the DEM data (EPSG 3035). Due to the temporal resolution of the simulation data of 3 hours being relatively low in relation to the total time period of the visualised precipitation events of 60 hours, the data between two consecutive time steps were linearly interpolated on a cell-by-cell basis and temporal resolution was increased by a factor of 10 with the interpolated data. This allowed for a smoother transition in the visualisation process. The linearly interpolated precipitation data were then stored in Wavefront obj format. For this, a similar procedure as for the DEM data was used but without domain splitting due to the coarser spatial resolution of the simulation data.

We imported the raw 3D data to Blender, where we reduced the number of vertices of each time step by 90% with the help of the Decimate function. Additionally, we iteratively applied the Laplacian Smooth Operator in order to polish the data by removing jagged edges. Finally, we exported each time step as individual 3D object.

3. Implementation of the visualization with the Unreal Engine

With all 3D models being ready for use, we imported the texturized DEM subdomains and all time steps of the simulation in Unreal Engine 4[8]. In order to enhance the visibility of the elevation changes as well as the rainfall data, we multiplied their elevation values by 3 and 25, respectively. Due to the sheer data size and the resulting rendering time, we equipped the DEM with a level of detail system, which temporarily substitutes far away subdomains with less detailed versions.

[7] https://CRAN.R-project.org/package=ncdf4

[8] https://www.unrealengine.com/en-US/what-is-unreal-engine-4

3.1 Data display

During the application of the preprocessed data, we strove to provide prospective users with an environment that is as familiar and easy to grasp as possible. Thus, we marked each of the more prominent cities with a floating nameplate. These nameplates adjust their rotation to permanently face the position of the head-mounted display.

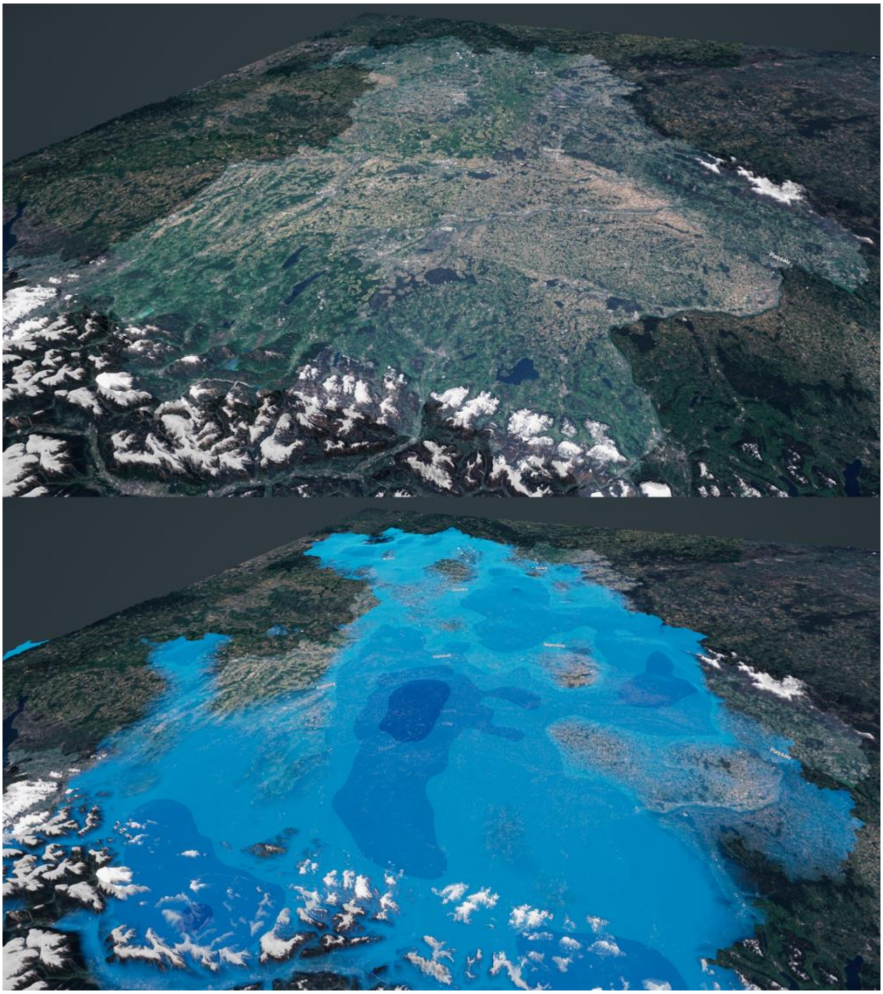

Fig. 1. Top: The digital elevation model of Bavaria texturized with satellite pictures of Bavaria. The geographical aspects of Bavaria are colored brighter to ease identification of borders. Bottom: Accumulated precipitation data are added to the visualization. Darker hues, increased opacity and higher elevation of data points signify stronger rainfall.

This keeps the names legible, irrespective of one's position and flight path. This small feature greatly increases the users' ability to orient themselves and understand the geographical context. This holds especially true for users without previous geographical in-depth knowledge of the presented section of the Earth. To further ease the comprehension of the digital elevation model, we increased the color intensity of regions within the borders of Bavaria, which can be seen in the top image of Fig. 1.

For the coloration of the data of accumulated rainfall, we opted against the use of linear colors. Instead, we separated the data along isolines and dyed the areas between these contours uniformly. This allows for a quicker categorization and comparison of different data peaks. In addition to the color value, we also visualize the data by means of elevation and opacity. Thus, the user can still scrutinize the full level of resolution of the data with the help of the redundant display techniques. Drawing the data with partial opacity also bears the advantage that the user can discern the geographical context, even though the rainfall data cover large parts of the surface, as can be seen in the bottom image of Fig. 1.

3.2 User interaction

In order to not obfuscate the user with too many buttons and additional displays, we limited ourselves to one of the two Vive Controllers. Apart from physically moving through the real room, which also translates to movement in the virtual world, the user is able to fly in any desired direction by the push of a button: The trigger on the backside of the controller regulates the movement speed, while the direction of movement is determined by the direction in which the user is holding the controller. With this, we aim to keep the navigation within the virtual scene soft, intuitive and lightweight. Fig. 2 shows the graphical user interface.

The controller also serves as the central access point for the steering of the simulation. The circular trackpad of the Vive controller offers four buttons for assignment of functions; one button at 12 o'clock. 3o'clock, 6 o'clock and 9 o'clock, respectively. Button input at the 12 o'clock position alternates between playing and pausing the simulation. A single press at the 3 o'clock position skips ahead to the next precipitation event, while a continued press fast-forwards through the currently displayed event. Analogously, a single press at the 9 o'clock position switches to the first time step of the event or, if already at the beginning, to the previous event. A long, continued button down event at the 9 o'clock position enables the user to fast backward through the event. The button at the 6 o'clock position, on the other hand, switches between the first and last time step with each press. This allows for an easy comparison between the final extents of each rainfall event without having to watch the whole event unfold. As wearing a head-mounted display prevents the user from seeing their finger positions on the controller, we added a small blue orb, which shows the current position of the thumb on the trackpad. Furthermore, we attached an icon to each of the four trackpad positions to represent their function. Each respective icon is highlighted blue if the user's thumb resides within the icon's quadrant and a press would trigger the function of the corresponding button.

We also included a visual display, which informs the user about the timeline of the rainfall event. The time and date of the currently displayed time step are shown at the top of the controller. We also attached a legend explaining the meaning of the color values to the left of the controller.

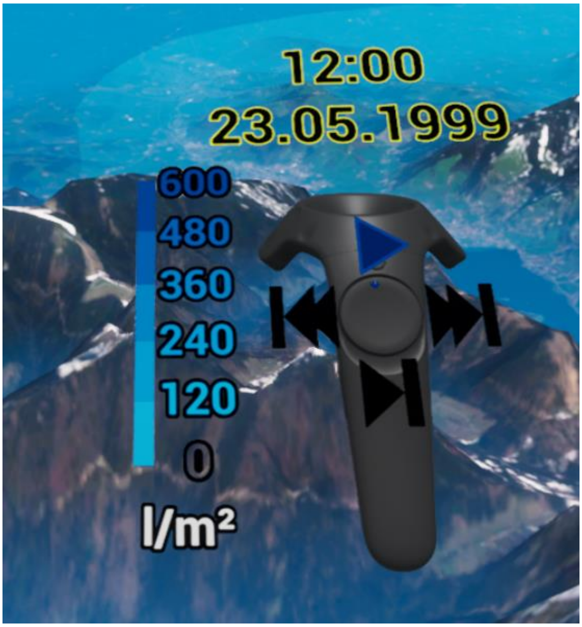

Fig. 2. View of the GUI attached to the Vive Controller. Atop the controller is a display of the current time step of the active rainfall event. To the left of the controller is a legend for the different hues of rainfall intensity. Along the circular trackpad of the controller are icons, which represent the different interaction functions that can be called by button press. Additionally, we added a small blue ball, which tracks the thumb position, to ease selection. The would-be selection is also highlighted blue.

4. Results and future work

During the first tests of our visualization, we found two characteristics shared among the interaction behavior of testers, which were previously not familiar with this visualization. First, our users were able to easily adopt the user interface and fully interact with the scene within the first 30 seconds after a few pointers. This applies to both digital immigrants as well as digital natives, although the latter adapted noticeably faster, which is to be expected. The second noteworthy behavior aspect was the impulse to at first ignore the displayed scientific data and inspect the geographical map. Most users begun their exploration of the virtual world by trying to find their home and other locations they visited previously in the real world. Another popular activity, irrespective of a user’s age or standing, were flights through the Alpine valleys as close to the ground as possible while trying to avoid the mountains. After sating these playful impulses, the users were interested on focusing on the scientific data.

In summary, we presented the workflow, and its working implementation, to visualize scientific climate data in immersive virtual reality. We also detailed an interaction design to intuitively explore scientific simulations. Further interesting research topics include a thorough user study to evaluate the interaction design as well as the application of this workflow to other climate attributes or geographical regions.

Acknowledgements

We would like to thank Elisabeth Mayer, Lea Weil and Kris Weinand for their invaluable help and support with processing of 3D objects as well as beneficial discussion and feedback. We also gratefully acknowledge funding of the project ClimEx by the Bavarian State Ministry of the Environment and Consumer Protection.

References

[1] Sweeney, T (2015). If You Love Something, Set It Free. [accessed July 26 2018]. https://www.unrealengine.com/en-US/blog/ue4-is-free.

[2] Wiedemann, M., Anthes, C., Bunge, H. P., Schuberth, B. S., & Kranzlmüller, D. (2015). Transforming Geodata for Immersive Visualisation. In: *2015 IEEE 11th International Conference on e-Science* 249-254.

[3] Knabb, K. A., Schulze, J. P., Kuester, F., DeFanti, T. A., & Levy, T. E. (2014). Scientific visualization, 3D immersive virtual reality environments, and archaeology in Jordan and the Near East. In: *Near Eastern Archaeology (NEA)*, *77*(3), 228-232.

[4] Wurm, L., García, R., Anthes, C., Kranzlmüller, D., & Höhl, W. (2016). Benefits of tablet interfaces for immersive visualization in information visualization.

[5] Berg, L. P., & Vance, J. M. (2017). Industry use of virtual reality in product design and manufacturing: a survey. *Virtual Reality*, *21*(1), 1-17.

[6] Helbig, C., Bauer, H. S., Rink, K., Wulfmeyer, V., Frank, M., & Kolditz, O. (2014). Concept and workflow for 3D visualization of atmospheric data in a virtual reality environment for analytical approaches. In: *Environmental earth sciences*, *72*(10), 3767-3780.

[7] ClimEx Project. [accessed July 26 2018]. http://www.climex-project.org/?language=en

[8] European Environment Agency (2017). Copernicus Land Monitoring Service - EU-DEM. [accessed July 26 2018]. https://www.eea.europa.eu/data-and-maps/data/copernicus-land-monitoring-service-eu-dem

[9] Roger Bivand, Tim Keitt and Barry Rowlingson (2017). rgdal: Bindings for the 'Geospatial' Data Abstraction Library. R package version 1.2-16. [accessed July 26 2018]. https://CRAN.R-project.org/package=rgdal.

[10] Landesamt für Digitalisierung, Breitband und Vermessung (2016). OpenData. [accessed July 26 2018]. https://www.ldbv.bayern.de/produkte/weitere/opendata.html