Otto-von-Guericke-University Magdeburg
Faculty of Computer Science
Dep. of Simulation and Graphics

German Aerospace Center Braunschweig
Institute of Simulation and Software Technology
Dep. of Software for Space Systems
and Interactive Visualization

Diploma Thesis

Adding Haptic Feedback to Geodesy Analysis Tools used in Planetary Surface Exploration

April 22, 2014

Author: Ronny Wegener[1]
Area of Study: Computer Science
Mat. No.: 162974

Supervisors Prof. Dr.-Ing. Bernhard Preim[1]
Dr. Robin Wolff[2]

Advisors: M.Sc. Antje Hübler[1]

[1]Otto-von-Guericke-University Magdeburg
[2]German Aerospace Center Braunschweig

Abstract

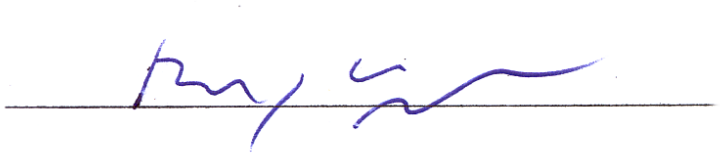
Force feedback supported user interaction is increasingly finding its way into various professional applications and areas of research. Tentative steps in the development of haptic supported Geographic Information Systems (GISs) led to the conclusion that force feedback has the capability to improve the user experience. This work was focused on the investigation of this assumption in the context of planetary surface analysis. For this purpose an existing planetary terrain visualization and exploration framework was extended by a haptic interface consisting of a haptic device and a haptic rendering pipeline. A selection of commercial devices was reviewed and assessed regarding to their suitability for particular terrain analysis tools. A challenge was to find suitable force feedback related algorithms to tackle the massive amount of planetary terrain data and adapt them into the haptic rendering pipeline. Two geodesy analysis tools have been enhanced with virtual fixtures to assist in their operation. A final pilot user study was conducted to compare the usability of the prototype haptic interface implementation with the usability of the original terrain visualization and exploration framework.

The haptic rendering pipeline and the selected haptic device were successfully integrated in the existing framework and operated with a proper refresh rate. The results of the measurements and the user feedback regarding the comparison of both systems were balanced. The initial assumption that force feedback will improve the workflow of planetary terrain exploration and analysis could not be confirmed. Some weaknesses were exposed in the haptic supported prototype and also in the original framework. However, some of the measurements and the user reports indicated that the workflow can still benefit from force feedback after a future revision to eliminate the weaknesses.

Declaration of Authenticity

I, Ronny Wegener, declare that the work[1] presented here is, to the best of my knowledge and belief, original and the result of my own investigations, except where indicated otherwise. This exegesis contains no material that has been submitted previously, in whole or in part, for the award of any other academic degree or diploma.

Magdeburg April 22, 2014



Signature

[1] PDF File
Web-View: `http://goo.gl/7O2K82`
Download: `http://goo.gl/7EPPGH`

Contents

List of Figures

Abbreviations

DLR	German Aerospace Center
NASA	National Aeronautics and Space Administration
ESA	European Space Agency
MGS	Mars Global Surveyor
MOLA	Mars Orbiter Laser Altimeter
HRSC	High-Resolution Stereo Camera
DTM	Digital Terrain Model
DEM	Digital Elevation Model
HEALPix	Hierarchical Equal Area isoLatitude Pixelisation
GIS	Geographic Information System
API	Application Programming Interface
VR	Virtual Reality
MarsVis	Virtual Reality terrain visualization and exploration framework
HCI	Human–Computer Interaction
DoF	Degrees of Freedom
GUI	Graphical User Interface

1 Introduction

In planetary research the interactive exploration and analysis of the surface plays an essential role to determine geodesy properties, make conclusions for geologic processes and find clues about the origin and the evolution of planetary bodies. Such information can be acquired by thorough spatial analysis and interpretation of topographic data and terrain features such as rifts in the crust, volcanoes, impact craters, sedimentation, or traces of erosion and material transport. Particular similarities between Mars and Earth plus the passable distance makes the neighbor sibling an interesting place for such kind of studies. Respective terrain data was gathered during the last decades through the Mars Global Surveyor (MGS) mission by the National Aeronautics and Space Administration (NASA) and the Mars Express mission by the European Space Agency (ESA). A convenient way to examine the surface based on the acquired terrain data, is to use a Geographic Information System (GIS) with appropriate software tools for geodesy analysis. The German Aerospace Center (DLR) is using an inhouse developed Virtual Reality terrain visualization and exploration framework (MarsVis), a GIS customized for planetary exploration and analysis (e.g. the Mars).

1.1 Research Objective

Currently the interaction between a user and the MarsVis is solely based on visual feedback. The here presented work arose as a straight sequel of the MarsVis project [35], with the purpose to find a way to enhance selected geodesy analysis tools with force feedback. Furthermore the research objective of this work is to examine whether force feedback can improve the utilization of the implemented geodesy tools. Two tools will be focused for this objective, the *profile liner* and the *volume measurer*. One goal is to deliver a realistic contact feeling every time when the user controlled tool collides with the surface. The user should be able to sweep over the terrain while feeling the shape and discontinuities at the same time. In this context appropriate data structures and haptic algorithms need to be investigated which will be convenient for a particular Digital Terrain Model (DTM). The realistic feeling when sweeping over the surface will be complemented by adding suitable *virtual fixtures* (haptic assistant and guiding techniques), such as vibration, snapping or repulsion. The fixtures will be chosen and customized depending on the needs and specifications of the planetary researchers. In a final conducted user study the workflow of the MarsVis with the force feedback assisted geodesy analysis tools will be investigated and evaluated in comparison to the original system without haptic support. This evaluation is expected to approve the concepts and verify the benefits of the profile liner and volume measurer from force feedback.

1.2 Motivation

In the last decades haptic interfaces became quite popular in various areas such as medical simulation, tele-operation, 3D modeling and entertainment. Current GISs are still relying on visual feedback only, which makes it difficult to inspect hidden areas, or detect certain topographic features such as rims or other surface discontinuities. Regarding the success of force feedback in other areas of study, an improvement of the intuitiveness and precision of haptic driven geodesy analysis tools can be expected.

Using haptic devices in conjunction with GISs is still in its infancy. Though, there are already a few related publications [10, 11] in this field of research, which will be introduced in section 2.4. However, none of them is covering the main objective of this work. The tools and constraints investigated in these papers are not convenient to be applied to the geodesy analysis tools because the characteristics of the described tools differs partially from the needs of the profile liner and the volume measurer. Furthermore these existing implementations are limited to a restricted area and viewport where they deal with convenient small surface patches. In contrast the MarsVis provides a seamless navigation through a tremendous amount of terrain data. Handling this huge surface requires different underlying implementation strategies. The workflow and behavior can also be influenced when operating in a free environment instead of operating in a restricted area. As a result of all that, the conclusions about the improvements and benefits of force feedback from related findings, can not be validated for the realization of this work. An independent and adjusted evaluation of this implementation is necessary to verify the successful improvements.

1.3 Challenges

Mars Express and MGS, both missions provide a vast amount of raw planetary surface data for further processing and spatial analysis. At the time of writing the compiled multi-resolution DTM of the MarsVis has a size of around 120 GB and is still incomplete, as long as the Mars Express mission is ongoing and incoming data still being processed. In comparison an dataset from the moon with a size of 2.2 TB is available at the DLR. The ability to explore the whole planet, where the user can navigate to arbitrary places, freely scale the distance above the ground level and changing the viewport, leads to an environment with a high dynamic workflow. The implemented structures and procedures for haptic rendering must be capable of accessing and extracting local samples from the large dataset, performing the haptic calculations and delivering the resulting forces at sufficient update rates. A refresh rate that is high enough to provide the feeling of stiff object such as the surface of a planet is essential. A particular frequency depends on the context and can not be addressed yet, but unrelated documents often refer to a rate of at least

1 kHz to represent stiff objects. Another problem that arises from the dynamic interaction, are the frequent changes of the spectator's position and viewport. The mapping between the workspace of the haptic device and the virtual space must be updated whenever the viewport changes. To determine the mapping parameters, the reachable area of the haptic device must be fitted effectively to the visible region. An additional difficulty with respect to the main research objective, is the design of a virtual fixture to assist the user when placing the control points of the poly-line for the volume measurer along certain terrain features such as mountain rims or peaks. To the best of our knowledge, there is no virtual fixture that considers multiple influences, e.g. five mountain peaks and the mutual interaction of their corresponding attracting forces.

1.4 Thesis Outline

Chapter 1 (Introduction): The introduction was the first of six chapters in this document.

Chapter 2 (Fundamentals): The next chapter will introduce some fundamentals and mentions related work to facilitate a better understanding of the subsequent parts. It starts with an overview of the basics such as GISs. Furthermore the general importance of the sense of touch in computer science related applications will be exposed. An insight into other disciplines will be given to show the success of additional haptic integration and justify the assumption, that haptic support can also improve the usage of geodesy analysis tools. Addressing the state of the art technologies will complete this part.

Chapter 3 (Concept and Design): The most comprehensive chapter contains all the necessary steps, to prepare the integration of a haptic device into the existing visualization framework. It starts with the investigation and assessment of appropriate haptic devices. A convenient *haptic terrain model* will be designed to supplement the existing DTM. Furthermore a strategy for dynamic mapping between the workspace of the haptic device and the the virtual space, will be explained. After discussing the concepts of virtual fixtures for geodesy analysis tools, the composed force rendering pipeline will be presented. This includes the explanation of the customized algorithms such as collision detection, force response calculation and virtual fixture processing.

Chapter 4 (Implementation): After discussing the concepts and designs, the technical details to realize a prototype from the developed functionalities will be justified. In the first place this will include details about the module structure for the integration into the existing visualization framework. Additional libraries are mentioned as required, along with the haptic device driver and the Application Programming Interface (API) for the device integration.

Chapter 5 (Evaluation): This chapter is focused on a pilot study to validate the developed prototype and examine the workflow benefits from the additional force feedback assistance. The aim is to provide meaningful test results for a comparison between the non-force and force supported usage of the selected geodesy analysis tools. Points of interests are the technical background of the participants, the design of the tasks that should be performed, the selection of the measured data and the the creation of a questionnaire to gather and evaluate subjective impressions.

Chapter 6 (Conclusion): The work will close with a summary of the developed prototype and the presented results from the user study. Existing problems will be reviewed and discussed, along with addressing possible improvements for future work.

2 Fundamentals

This chapter will explain essential background knowledge in the context of the research objective and is focused on geographic information systems, haptic devices, haptic applications and haptic rendering. It will also cover the state of affairs of prototypes for terrain exploration and analysis that already facilitates force feedback.

2.1 Geographic Information Systems

The objective of Geographic Information Systems (GISs) is to process and work with geographic data and related information. They consist of multiple components that interact among each other. A detailed introduction can be found in the GIS primer [12]. There is no specific definition, but the following explanation comes very close to summarize the term GIS.

> "An integrated collection of computer software and data used to view and manage information about geographic places, analyze spatial relationships, and model spatial processes. A GIS provides a framework for gathering and organizing spatial data and related information so that it can be displayed and analyzed." [15]

Derived from this quote, the main purpose of a GIS is to organize, analyze and visualize descriptive information that is related to geographic locations, e.g. roads, cities, population dense and more. Such information is usually represented as points, lines, polygons and annotations or a mixed set of them and managed in either two dimensional raster or vector layers. Nowadays such systems can be operated on a classical desktop environment, where the terrain is usually projected in plan view (seen from top) onto the screen. A mouse or a pen digitizer is mostly used as pointing device for user interaction. More advanced GISs, such as ArcGIS, offer exploration in a three-dimensional view of the terrain, but those are currently missing the ability of further interaction beyond navigation.

Tools

Tools are part of the software component of a GIS and are important to perform exploration and analysis tasks, or manipulate the existing data. They originated in the past where paper maps were still used for information retrieval. Pens, rulers, triangles and dividers are just a few of them. Today's tools are software solutions embedded in GISs to perform sophisticated processing of the digital information. Besides customized tools for specialized applications, there are also general tools for common techniques. Such core tools can manipulate point(s), line(s), polygon(s) and geometry collection(s) and link them to the related information, e.g. draw a poly-line which will present a road and link it to a speed limit. Also tools to edit text based annotations belongs

to the basic stock. Presentation tools can perform operations to modify the information that will be viewed. Common operations are buffering, clipping, dissolving, intersection, merging and unification of data.

2.2 Digital Terrain Models

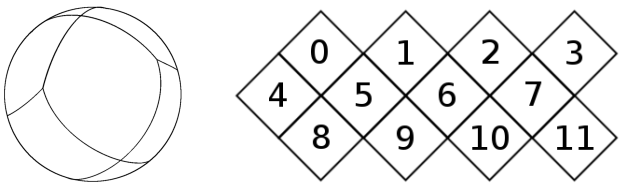
A Digital Terrain Model (DTM), also referred to as Digital Elevation Model (DEM), is a spatial representation of the sole ground level for solid planets or parts of them. Any objects such as building or plants are excluded, water or ice can be added optionally depending on the application. The underlying data can be structured as a raster with the corresponding height values for each raster point (heightmaps), or as a network of connected vertices (polygon mesh). In terms of GISs it is a set of locations, where each point is connected to the corresponding height or elevation value. The significant information can be acquired and retrieved through various instruments and techniques. The terrain data of the Mars was sampled with the Mars Orbiter Laser Altimeter (MOLA) and with a High-Resolution Stereo Camera (HRSC).

2.2.1 Spherical Coordinates

A common presentation of spherical coordinates is by using *latitude* and *longitude* values. Both parameters are angular. Assuming the z-axis goes through the north and south pole of a planet, than the angle between the z-axis and a ray from a point on the sphere to the origin is called latitude. The angle in the projection plane perpendicular to the z-axis is the longitude. A zero meridian indicates where the longitude is zero.

2.2.2 HEALPix

The Hierarchical Equal Area isoLatitude Pixelisation (HEALPix) [32] technique is a map projection that divides the surface of a sphere into pixels or patches. All patches are curve-linear, quadrilateral and covers exactly the same size of area which is convenient to map them on squares. In contrast to other map projections, there is no distortion near the poles. The parametrized u-v representation of a patch has the advantage of independence from absolute ecliptic or Cartesian coordinates. The smallest possible sub-division of a sphere is 12 base patches as shown in figure 1.. To split the sphere into the next higher number of patches, each base patch is divided into four equal sub-patches, resulting in 48 patches.



(a) Sphere view (b) Map view

Figure 1: HEALPix grid with 12 base patches

2.2.3 Quad-Trees

A quad-tree is a data structure that consists of nodes. The root node is the origin of the tree and has no parent node and four child nodes. Each internal node has a parent node and also four children. The lowest level nodes that only have a parent node, but no children, are called leafs. Quad-trees can be used to organize and query data that is related to a quarter based subdivision scheme such as partitioned two dimensional areas. They are suitable for HEALPix grids, where each patch can be divided into four children.

2.3 Haptics

A human uses five senses to recognize and interact with the real world environment. Advanced research in the field of Human–Computer Interaction (HCI) investigates the usage of these senses to facilitate an intuitive and effective interaction between humans and computers. Haptics in the field of computer science and technology is the study of user interfaces that take advantage of the sense of touch. The sense of touch can be classified into two categories regarding the type of their recognition. *Cutaneous* or *tactile feedback* is everything that can be felt with the skin, such as roughness, friction or micro vibrations. The ability to perceive motion and positioning of joints and limbs by receptors in the muscles is called *kinesthetic* or *force feedback*. Including the sense of touch in HCI can be beneficial to various different applications, especially when interacting with virtual physical objects. Further information can be found in the second chapter of the book "HCI Beyond the GUI" [1], which is focused on an introduction to haptic interfaces.

2.3.1 Haptic Hardware

Hardware that is capable to display forces or stimulate the sense of touch is called a haptic display or a haptic device. Haptic displays can basically be classified into two categories. Depending on the working method a haptic device can be either assigned to the category of *tactile devices* or to the category of *kinesthetic devices*. There is also the possibility to combine devices of both types to utilize their advantages.

Tactile Devices

Tactile devices are focused to convey the sense of touch that is perceived through the skin. They are appropriate to simulate the feeling of physical properties such as pressure, friction and roughness. Usually such devices are attached to the area of the skin that should be stimulated. The fingertips have the highest density of receptors and are the most sensible part of the skin. Also the hands and fingers are the most important commodity when interacting with objects in an environment. As a result of this, tactile devices are mounted and operates on the hands and fingers. Sophisticated tactile devices are still subject of thorough research and not commercially available.

Braille displays are flat surfaces consisting of an array of small pins. The pins can be displaced to stimulate the skin through pressure. Vibration-based displays also belongs to the mechanical solutions, Electric motors or piezoelectric elements generate a periodical force by oscillating a mass with a given frequency. There are also some state-of-the-art devices that are based on different approaches such as electrostatic modulated friction [25], electrical stimulation of the receptors in the skin [26], radiation pressure of ultrasound [27] and generated air vortex rings [28].

Kinesthetic Devices

Kinesthetic devices, also referred to as force feedback devices, are targeting the muscles and joints. Usually the focus lies on the most moving and sensing parts of the human body, the fingers, the wrists and the limbs. These devices need to be anchored to a ground that is independent of the involved joints. A handle also referred to as *end effector* is the element directly controlled by the user and is responsible for the interaction between the user and the device. Depending on the application, an end effector can have different shapes and additional input elements such as triggers or buttons. Kinesthetic devices can follow one of two different principles. Admittance control based devices utilize sensors to measure the forces applied by the user. Depending on the incoming force, a calculation will be performed and shift the user to a new position. The second principle is known as *impedance control*. Sensors are embedded into the device to determine the position and the orientation. Forces and torques will be calculated depending on the sensor input and delivered to the user.

The most common and commercially distributed devices are *motorized linkage devices*, see Figure 2. Both, the mount point and the end effector are connected through a system of mechanical arms with joints. The joints consist of electrical actuators to apply forces and torques to the end effector.

(a) Geomagic Touch (Phantom Omni) (b) Force Dimension Omega 6

Figure 2: Low-cost kinesthetic linkage haptic devices

Actuated cable solutions [30] utilize an end effector that is connected to multiple cables. The cables are mounted on a frame. A force is applied by manipulating the tension of the cables. Displays with an end effector that levitates in an electromagnetic field [29] also belongs to the family of kinesthetic devices.

2.3.2 Haptic Rendering

When graphic rendering is the process of creating pictures from a virtual model and audio rendering is the process of creating sound from a virtual model, then haptic rendering can be described as the process of creating forces from a virtual model. Kenneth Salisbury et al. did a lot of research in the field of haptic in computer science and [16] explains briefly the general concepts, discusses the force rendering and gives an overview of the main components for a haptic rendering pipeline. The base layout of the rendering pipeline, as shown in Figure 3, is divided into three main blocks.

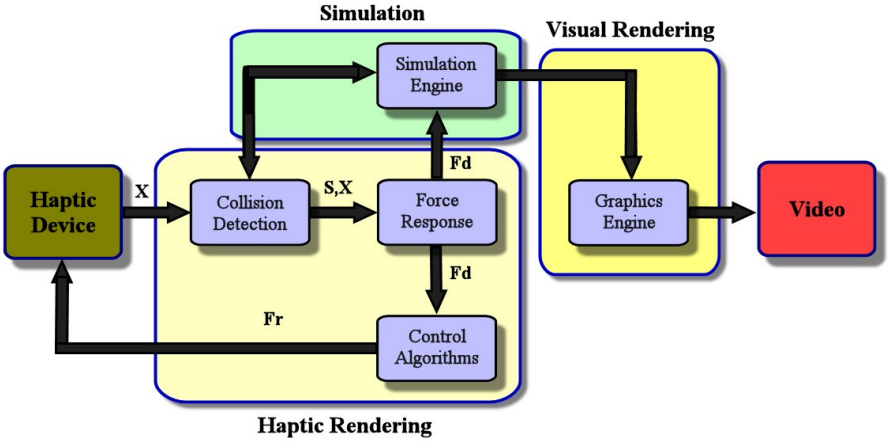


Figure 3: Haptic rendering components (Source: [16])

The collision detection algorithm is responsible to check for contacts between the *avatar* (virtual presentation for the user interaction) and the objects in the

simulation model. Depending on the results from the collision detection, the force response algorithm calculates an interaction force between the avatar and the colliding virtual objects. The last block contains an algorithm to adjust the output of the force depending on the context of the virtual environment and the capabilities of the output device.

Coordinate Mapping

Usually the coordinates in the *workspace of the haptic device* need to be transformed to the corresponding coordinates in the *virtual space* of the simulation model. This can be done by a mapping function that applies a transformation matrix to translate, rotate and scale the coordinates. Sometimes the haptic workspace can only cover a small area of interest in the virtual environment. In [23] a solution with a drifting workspace is presented to solve this problem. Basically the mapping of the workspace will be relocated to follow the current position of the avatar. This approach is based on the fact, that slight workspace drifts are not perceived without the corresponding visual feedback while the hand is under motion. Workspace mapping in the context of collaborative work considering mixed haptic devices is discussed in [24].

Object Collision

A collision between two or more virtual objects occurs, when they have contact in at least one point, or to put it another way, two or more objects collide when they intersect. The *penetration depth* is the minimal distance required to separate two colliding objects.

Hooke's Law

When two virtual objects collide, a force occurs between these objects. The force is modeled after Hooke's law based on the expansion of a spring. The displacement and the force are related linearly. The constant factor depends on the physical properties of the spring and is called *stiffness*.

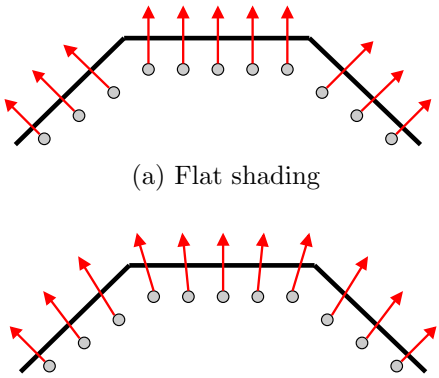
Force Computation

There are plenty of algorithms available to calculate the resulting force. The choice of the algorithm depends on the context of the simulation model. Assuming simple geometric objects, a vector field depending on the avatar's location and the penetration depth can be used. The *god-object* paradigm [18] improves the vector field method by using a modified avatar position, which is optimized between the surface of the penetrated object and the penetration depth. More sophisticated solutions are focused on calculating the collision of complex volumetric objects. McNeely et al. [19] developed an algorithm that discretized each object into a set of voxels and a set of points covering

the outer shell. An existing collision between two objects can be detected by testing all points from the shell of the first object against all voxels from the second object. Other approaches utilized hierarchical bounding volumes to describe the objects. When the bounding volumes of two objects intersects, a collision between those objects can be assumed, or an extended collision detection algorithm based on the bounding volumes' intersection area can be started.

Force Shading

When using polygons, the computed force points in the same direction as the normal vector of the polygon. This can be compared to flat shading in computer graphics. Figure 4 illustrates the problem of resulting force discontinuities in a polygon mesh, which is also addressed in [21, 22]. Consequently the force direction changes abruptly when moving over a polygon edge and this conveys the feeling of a cracked surface. Shading techniques from computer graphics can be applied to smooth the force directions over the edges and solve this problem.



(a) Flat shading

(b) Enhanced shading

Figure 4: Shading of responding forces

Haptic Textures

Textures are another concept from computer graphics, that can be adapted to the haptic domain as well. They can be used to map material properties such as stiffness, roughness or friction which affects the resulting force vector. Besides those material properties, any other haptic related characteristics can be used to modify the force vector. Textures are also a convenient way to alter the appearance of the surface e.g. by bump mapping or displacement mapping. Further information about haptic textures can be found in [22].

Virtual Fixtures

Virtual fixtures, sometimes called haptic fixtures or haptic constraints, are a metaphor for assisting and guiding users with fictional constraints. This paradigm [5] was introduced in 1993 by L. B. Rosenberg. An example for a real world fixture can be found in the process of drawing a line between two points on a sheet of paper. Moving the pen freehand from one point to the other results in a wavy line. Taking a ruler to help in the drawing process will produce a straight line. In this case the ruler can be denoted as a fixture. Some examples for virtual fixtures in the haptic domain are vibrations to note or to warn the user, snapping to points of interest or repelling from restricted areas.

2.3.3 Haptic Applications

In various studies [2, 3, 4] the effects of stimulating different senses were compared. It was revealed that the sense of touch is as relevant as the other senses (sight and hearing), which gained well paid attention in computer science. The response time of touch based stimulation was even slightly better than reactions on visual or audible stimulation. This is an example that the user experience can benefit from involving additional senses, in this particular case the ability to feel forces. This has also been justified in various tests conducted over the last years, e.g. an early user performance experiment with haptic supported force feedback [8], or by a recent sculpting and haptic constraints evaluation [6].

Force feedback based Human–Computer Interaction already found its way into various application areas. Haptic support had a high impact in medical simulation and gained a noticeable significance in remote controlled surgery assistant systems (Figure 5). A survey of medical training simulators [7] gives a detailed overview of the various capabilities and their application. Besides remote surgery, force feedback assistance is widely used for tele-operation in robotics [9], e.g. to control machines in unreachable or hostile environments. Many other application areas such as 3D modeling or entertainment are also utilizing force feedback devices.

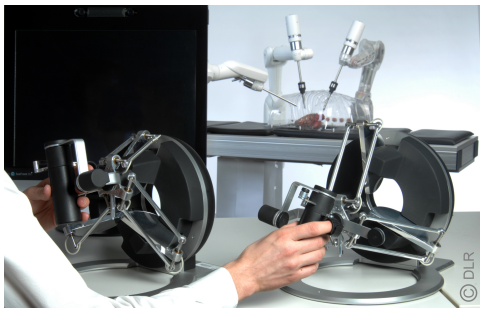


Figure 5: MIRO surgery arms controlled by Force Dimension haptic devices

2.4 Related Work

2.4.1 Haptic for Terrain Processing

The success of force feedback in various disciplines yields a lot of knowledge and leads to the assumption, that force feedback can also improve geoscientific tools and is worth further investigation. First attempts were already made to integrate force feedback into GIS prototypes.

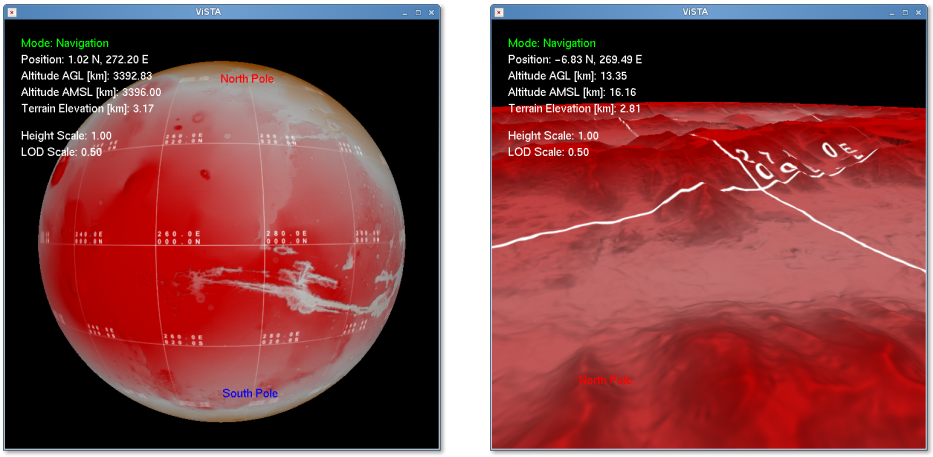
The M4-Geo framework [10] was an implementation to combine common manipulation tasks performed on geospatial data with force feedback driven user interaction. All operations were exclusively performed with the haptic device, including navigation and Graphical User Interface (GUI) interaction. The tools presented in this work included a pen to draw lines and poly-lines on the terrain, a pressure based tool for circular and flooding based selection and a terrain deformation tool. A terrain region was extracted from an existing DTM and preprocessed with resolutions ranging from 2048×2048 to 4096×4096 raster points. A static mapping function was used to transform the prepared terrain region to fit in a 40 *cm* cube in workspace coordinates. The comments given by local domain experts after the evaluation of the framework were mainly positive and indicates the possibility of improvements by adding force feedback support to certain geoscientific related tasks.

A human-subject test, focusing on the evaluation and comparison of haptic and non-haptic driven methods to trace paths on a terrain, was conducted by Raghupathy et al. [11]. Three methods were tested in various setups. The first method was solely visual, without force feedback support. A drawing operation should occur when the pen reaches a threshold distance to the surface. For the next method a visual constraint was added. In case the surface was penetrated within a certain threshold, the visual presentation of the pen stays on the top of surface. The third method utilized force feedback support to prevent the user from penetrating the surface. The results have shown, that the method with force feedback was performed with higher accuracy than the other tested implementations and the completion time of the tasks was slightly decreased. Concurrently it has also been shown that visual constraints can improve the effectiveness of interaction without force feedback.

2.4.2 The VR Terrain Visualization and Exploration Framework

On behalf of the German Aerospace Center (DLR), a scientific Virtual Reality terrain visualization and exploration framework (MarsVis) [36] was developed to explorer and investigate the Mars in an immersive Virtual Reality (VR) environment. The framework also provides a set of selected tools [34] for spatial analysis. The MarsVis uses a 3D projection (Figure 6) instead of the more generalized 2D plan view (top projection) of common GIS. The focused development setup includes a powerwall (screen composed of multiple high-

resolution displays), shutter glasses for stereoscopic imaging, a tracking system for position detection and a fly-stick for navigation. However, the system has also the capability to be operated in a common desktop environment.



(a) 3392 km above ground (b) 13 km above ground

Figure 6: MarsVis screenshots

Digital Terrain Model

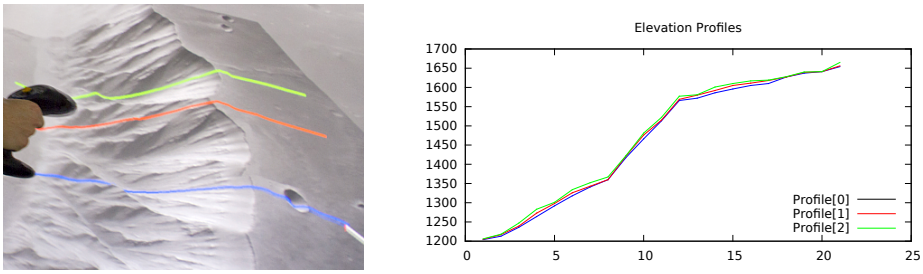
A preprocessed DTM is used to store and access the massive amount of terrain data. The underlying structure of the discrete DTM utilizes the HEALPix grid with a partition of twelve base patches. Each patch provides a resolution of 255×255 height samples. Further the HEALPix partitions are extended by a quad-tree scheme [33] to offer different level of details depending on the source data. Each of the twelve base patches are forming a root node of a tree. All four children of a node in a tree consist of a sub-patch that only covers a quarter of the area from the patch of the current node. The child nodes still share the same number of samples as their parent. Thus the composition of the four patches from the child nodes has a resolution that is four times higher than the resolution of the patch from the parent node. During the creation process of the DTM, all samples in each patch were interpolated from the original raw data, that was acquired through the Mars missions from the National Aeronautics and Space Administration (NASA) and the European Space Agency (ESA). The resulting quad-trees of the DTM for the Mars have a maximum depth of eight levels. In a forest of twelve exactly balanced trees this yields a total of 262140 nodes or patches. The maximum possible resolution at the highest level of detail is roughly $100\ m \times 100\ m$.

$$12 \times \sum_{l=1}^{8} 4^{l-1} = 262140 \tag{1}$$

The DTM is bijective, any kind of overhangs or caves will not occur in the terrain model. Each unique ray that is emitted from the origin of the planet, will intersect the surface in exactly one single point.

Geodesy Tools

Two important tools for geodesy analysis that have already been implemented in the MarsVis, are the *profile liner* and the *volume measurer*. The profile liner, shown in figure 7, plots elevation values which are sampled along one or more line segments. The user defines these line segments by placing their corresponding start- and endpoints on the surface.



(a) Profile liner (b) Plot comparison

Figure 7: Creating elevation profiles

The purpose of the second tool is to compute the volume enclosed by a user defined poly-line as illustrated in Figure 8. The zero surface of the poly-line and the surrounded terrain are forming the volume. The capacity of the volume is calculated from the integrated height differences of all the surface samples within the surrounded poly-line.

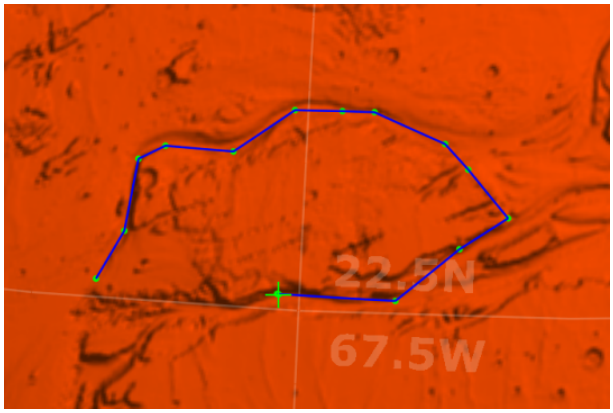


Figure 8: Bounding poly-line for volume measuring

2.5 Conclusion

This chapter gave an insight into Geographic Information Systems (GISs). Some related functionalities of the Virtual Reality terrain visualization and exploration framework (MarsVis), a GIS application that will be used for the implementation, were exposed. This includes the explanation of the Digital Terrain Model (DTM) structure and the operation of the two geodesy analysis tools: profile liner and volume measurer. Another integral part was the brief introduction into haptics for computer science. An overview of the different haptic device types showed the variety of available hardware. The software aspect was covered by explaining the essential rendering pipeline which computes a response force based on the user input and a simulation model. In addition the metaphor of virtual fixtures was illustrated with an example.

3 Concept and Design

In the context of this work it is necessary to discuss various different aspects and combine them to a complete solution to solve the research objective. Some of the major aspects are the selection of an appropriate haptic device, an efficient coordinate mapping between workspaces, the composition of a convenient haptic rendering pipeline and the design of virtual fixtures for the geodesy analysis tools. At the end it will be possible to translate the developed concepts into an implementation.

3.1 Haptic Device Assessment

Haptic devices vary in their usability and specifications. With an inconvenient device the benefit of additional haptic functionality can not be assured, so the choice must be considered carefully. A *Phantom Omni* haptic device is provided by the German Aerospace Center (DLR) for the realization of this project. In this section the applicability of the Phantom Omni as interface for virtual geodesy analysis will be examined and assessed. Despite the unavailability of other commercial devices for this project, they will be considered in this discussion as well.

3.1.1 Device Type

Standalone tactile displays can be excluded right from the start, since these are not capable to convey locational information corresponding to the virtual model. Tactile displays can not affect the muscles and joints which make them unsuitable to prevent a user from penetration or apply forces from virtual fixtures. On the other hand surface features, such as sharp edges, can be sensed clearly by stimulation of the skin.

Kinesthetic displays are much more suited to convey the feeling of virtual object boundaries. The advantage of admittance based kinesthetic devices is the ability to present hard surfaces with high stiffness. A drawback is the inability to deliver free movement. Impedance based kinesthetic devices allow non-resistant movement without limitations, but the presentation of high stiffness is a technical challenge for such devices. However, an impedance based kinesthetic device is more convenient since the stiffness limitation has a lower impact than the loss of the capability of free movement. A combination of a kinesthetic device and a tactile device has the advantages from both types.

From a commercial point of view, the variety of professional haptic devices is limited to kinesthetic devices. Tactile displays and displays with combined tactile and kinesthetic behavior are still immature and accordingly hard to find in the public market. Consequently a kinesthetic impedance device take the place as the current best choice until combined haptic devices are more

mature and commercially available. The following assessment of the device's properties is focused on kinesthetic impedance devices.

3.1.2 Device Properties

Size and Weight

The first properties that come to mind are the size and the weight. Both properties does not affect the usability regarding the research objective of this work. However, a portable desktop sized device is preferred. Without some exceptions most of the commercially available devices are suited for desktop usage.

Workspace Dimension

Another size that is much more essential is the dimension of the workspace where the user operates. A small workspace sacrifices fidelity and therefore accuracy. A large workspace might not be reached by the user through limitations in hand, wrist and arm movement. Operating within a much larger workspace over a longer period of time may also lead to fatigue of the user. The concrete size depends on subjective preferences and cannot be addressed. A personal recommendation is an effective workspace that covers a cube of roughly 20 cm. It is large enough to provide enough fidelity and can be reached without straining the wrists and limbs.

End Effector

The end effector is available in various shapes. Taking a look into the history of cartography, the pen and paper paradigm was consistent over the last centuries and is still common today. A pen shaped end effector seems most suitable as hardware interface to work effectively with the introduced geodesy analysis software tools.

Physical Resolution

The position of the end effector in the workspace is discrete and is determined by the resolution. A higher resolution provides a higher accuracy, but the absolute resolution depends also on the combination of the resolution and the effective workspace dimension. Mapping a virtual space to a small workspace with a high resolution can still provide less absolute discrete samples than a large workspace with a slightly lower resolution. The resolution of the device should at least fit the sensibility of the users hand and fingers. An estimated recommendation derived from other pointing devices, is a resolution of greater than 500 dpi, which is equal to $1/20$ mm.

Stiffness

The realistic representation of rigid objects depends on the stiffness offered by the device. The stiffness is a constant describing the change rate of force in dependence of the displacement. Displays with a higher stiffness are capable to convey the feeling of harder objects, which is expressly desired to work with high fidelity on planetary surfaces.

Force Range

Another property is the magnitude of the force that can be displayed. The device should be capable to deliver forces high enough to represent rigid objects and ensure the usability of appropriate virtual fixtures. The value depends on the users sensibility. Based on preliminary tests, the force difference for rigid objects between non-contact and full collision should be at least 2 N to be noticeable. For the virtual fixtures such as snapping, this value is also supposed to be more than sufficient.

Update Rate

Besides the force and the stiffness, there is a third property that affects the hardness of virtual objects, the update rate. The update rate determines how fast the device responds to a position change or how fast the force can be updated. Higher refresh rates reduces the time to detect a penetration and relay a responding force. Discussions about the boundary frequency to recognize hard objects are restricted to the range below 1 kHz and can be left out. All devices have an update rate equal or higher than this controversial scope and are capable to deliver a sufficient hardness feeling.

Degrees of Freedom (DoF)

Kinesthetic devices are bidirectional and have different DoF for the input and the output direction. The selected device must at least offer three input DoF to determine the current position in the workspace. Additional parameters such as pitch, roll and yaw are not necessarily required. Furthermore the device must be able to display the output force in three spatial dimensions in the workspace. A device with torque support is not necessary for the research objective.

3.1.3 Conclusion

Taking all these considerations into account, the Omega 6 haptic display by Force Dimension is theoretically the best choice. However, the provided Phantom Omni (recently known as Geomagic Touch) is an entry level haptic device and a cost-effective alternative to the Omega 6. All considered aspects are

almost satisfied by the Phantom Omni. It can be assumed that the Phantom Omni will be sufficient to implement and evaluate the concepts developed within this work.

Device	**Effector**	**DoF** In	**DoF** Out	**Workspace** [mm]	**Resol.** [mm]	**Force** [N]	**Stiffness** [N/mm]	**Rate** [kHz]
Geomagic								
Touch	Stylus	6	3	160 x 120 x 70	0.0550	3.3	1.02-2.31	1.0
Touch X	Stylus	6	3	160 x 120 x 120	0.0230	7.9	1.48-2.35	1.0
Premium 1.0	Stylus	6	3	254 x 178 x 127	0.0300	8.5	3.50	1.0
Premium 1.5	Stylus	6	3	381 x 267 x 191	0.0300	8.5	3.50	1.0
Premium 1.5/6	Stylus	6	6	381 x 267 x 191	0.0300	8.5	3.50	1.0
Premium 1.5 HF	Stylus	6	3	381 x 267 x 191	0.0070	37.5	3.50	1.0
Premium 1.5 HF/6	Stylus	6	6	381 x 267 x 191	0.0070	37.5	3.50	1.0
Premium 3.0	Stylus	6	3	838 x 584 x 406	0.0200	22.0	1.00	1.0
Premium 3.0/6	Stylus	6	6	838 x 584 x 406	0.0200	22.0	1.00	1.0
Novint								
Falcon	Knob	3	3	101 x 101 x 101	0.0635	8.9	N/A	N/A
Force Dimension								
Omega 3	Knob	3	3	160 x 160 x 110	0.0100	12.0	14.50	8.0
Omega 6	Pen-Shaped	6	3	160 x 160 x 110	0.0100	12.0	14.50	8.0
Omega 7	Gripper	7	3	160 x 160 x 110	0.0100	12.0	14.50	8.0
Delta 3	Knob	3	3	400 x 400 x 260	0.0200	20.0	14.50	8.0
Delta 6	Wristler	6	6	400 x 400 x 260	0.0200	20.0	14.50	8.0
Sigma 7	Gripper	7	7	190 x 190 x 130	0.0015	20.0	N/A	8.0
Haption								
Virtuose 3D Desk.	Stylus	6	3	200 x 200 x 200	N/A	3.0	1.00	1.0
Virtuose 6D Desk.	Stylus	6	6	200 x 200 x 200	0.0300	3.0	2.00	1.0
Virtuose 6D	Gripper	6	6	1300 x 658 x 1080	0.0200	10.0	2.00	1.0

Table 1: Comparsion of selected haptic devices

3.2 Haptic Terrain Model

The force computation depends almost exclusively on the interaction with the surface. Therefore the algorithms from the haptic rendering pipeline need extensive access to the underlying terrain. This includes direct access to the height value for a given location and its surrounding neighbors. The first thought is obviously to use the existing Digital Terrain Model (DTM) to access this data. However, the current DTM implementation was designed for visual representation and not for excessive data processing. Due to technical limitations such as stateless quad-tree traversal, missing neighbor iteration and the arrangement of different patches, the DTM cannot incorporate efficiently with the haptic rendering pipeline. Three strategies are considered to solve this implementation issue:

1. Redesign of the existing DTM to incorporate with the haptic rendering pipeline

2. Development of an interface between the DTM and the haptic rendering pipeline

3. Creation of a new model for the haptic rendering pipeline that coexists and synchronizes with the DTM

The current implementation of the HEALPix tessellated quad-tree based DTM is very complex. Modifying the DTM with respect to the haptic rendering

pipeline involves a thorough consideration of the present constraints to preserve the current behavior. From a technical point of view this represents a significant intervention which is time consuming and poses the risk to disrupt the reliability of the framework. Therefore a complete redesign of the DTM goes beyond the scope of the research objective of this work and is left out of consideration. The second option only simulates a convenient haptic model by translating the queries from the haptic rendering pipeline in requests that are implemented in the underlying DTM. Almost no modifications have to be applied to the existing model which simplifies the implementation and ensures the proper functionality. On the other hand the unchanged DTM in conjunction with an additional abstraction layer will generate an overhead, which is counterproductive in an attempt to improve the data acquisition. The last option is to complement the existing DTM with an additional haptic terrain model. Both models can be accessed independently. While the DTM remains unchanged, the haptic terrain model can be optimized for the interaction with the haptic rendering pipeline. The haptic terrain model will only present the current area of interest instead of the terrain from the whole planet to reduce the memory usage and improve the performance. A drawback of this solution is the required synchronization with the DTM whenever the area of interest changes.

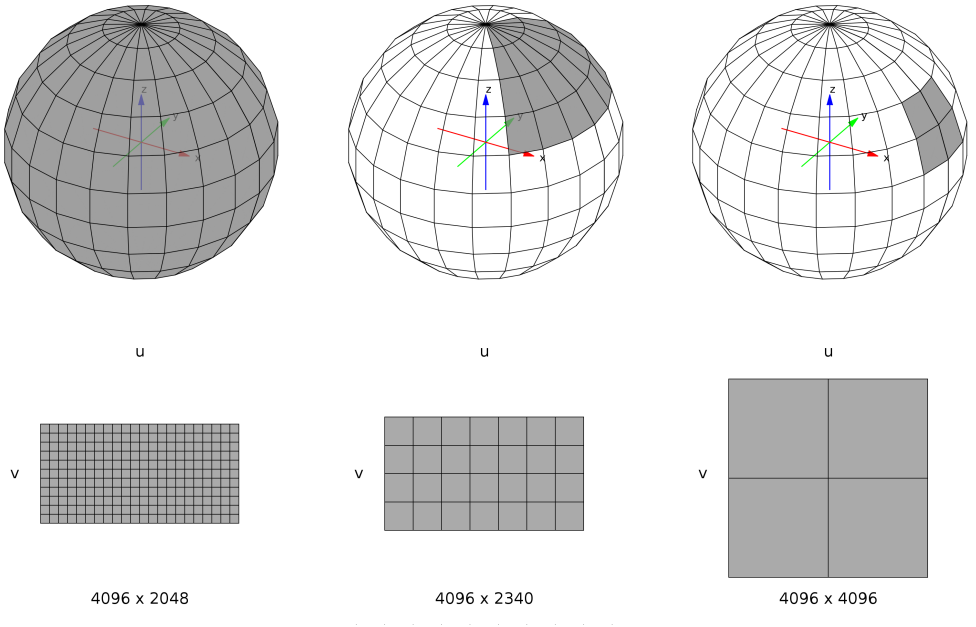
In favor of the performance the third strategy, which introduces an additional haptic terrain model, will be used. The synchronization issue can be ignored assuming that the user does not perform haptic interactions during a camera transformation. After the viewport has changed, the haptic terrain model is updated with the corresponding content from the DTM.

3.2.1 Data Structure

After the third strategy has been chosen, a particular structure to store the terrain data needs to be defined. A suitable way is to use a two dimensional parametrized *raster image* which is a common structure to represent geographic information e.g. *heightmaps*. The local parameters are labeled u and v. The index based access of a raster based structure is performant and allows the straightforward iteration through neighbored samples. Raster images consist of *equidistant* aligned pixels which leads to *distortions* when projecting coordinates *equirectangular* from a latitude and longitude enclosed region of the sphere to the raster image. The distortion caused by an equirectangular projection depends on the latitude and affects the horizontal distance between the pixels in the raster image. Some examples for equirectangular map projections of different regions on a sphere are shown in Figure 9.

There are plenty of other map projections available, which will not be discussed in detail. It is also a common approach to map the polar coordinates on other geometric objects such as cylinders or cubes. These primitives can

be unfold into two dimensional space to provide parametric access. All map projections have unique distortions which must be compensated. Unfolded geometric primitives might have gaps in the parametric representation which needs to be considered as well.



(d) Whole sphere (e) Large region (f) Small region

Figure 9: UV mapping of regions from a sphere to raster images

Size

The size of the static raster image depends on the maximum haptic workspace and the resolution. A bounding box for the Phantom Omni with the longest possible edge of 568 mm and a resolution of 0.055 mm, results in a recommended total number of roughly 10300 units per dimension. Referring to related work [10], the longer side of the raster image will be set to 4096 units to reduce the costs when processing the data. The assignment of the longer side and the value of the shorter side is determined by the latitude and longitude ratio spanned by the region of the sphere. The decreased resolution of the raster image will lead to a reduction of the available physical resolution for the Phantom Omni, which will drop in worst case from 0.055 mm to 0.14 mm per unit.

To compensate the non-linear deformation caused by the map projection in u direction, it is necessary to store the original length of each arc (A) that is covered by the corresponding row in the raster image. An arc (A) on the sphere is the part of the perimeter (P) between the minimum and maximum longitude (lg_{min}, lg_{max}). Furthermore the length of the arc (A_u) depends on

the latitude and the radius (R) of the planet. The radius (R) is also a linearly approximated function of the latitude (lt). The polar radius for the Mars is 3376 km, which is 20 km less than the equatorial radius.

$$R(lt) = 20 \cdot \sin(lt) + 3376 \tag{2}$$
$$P(lt) = 2\pi \cdot \sin(lt) \cdot R(lt) \tag{3}$$
$$A_u(lt) = \frac{lg_{max} - lg_{min}}{2\pi} P(lt) \tag{4}$$
$$A_u(lt) = (lg_{max} - lg_{min}) \cdot (20 \cdot \sin^2(lt) + 3376 \cdot \sin(lt)) \tag{5}$$

Analogous to the arc in u direction, the arc (A_v) in v direction between the minimum and maximum latitude (lt_{min}, lt_{max}) is also of interest. Fortunately the v direction is not affected by the deformation, the single arc (A_v) value is equal for each column. The different polar and equatorial radii involves numerical integration to calculate the arc length of an ellipse. As the difference between both radii is relatively low, the average radius $R = 3386\ km$ of both will be used as an approximation.

$$A_v = (lt_{max} - lt_{min}) \cdot 3386 \tag{6}$$

3.2.2 Parameters

At this point the haptic terrain model only contains information to map polar coordinates from a region on the sphere to the pixels in the raster image. This section will discuss which haptic relevant properties and values will be stored within the haptic terrain model. Some of the informations have to be globally available for the whole model, while other data will be stored for each raster point.

Height

The main purpose of this model is to store the height values for the selected region, so naturally the height will be stored for each raster point, see Figure 10. The height is the distance from the origin of the planet to the coordinate on the surface that corresponds to the raster point. There is also some statistical information about the selected region that should be available as global information. This includes values such as the minimum and maximum height, the height difference and the average height level. This information is used as reference for other computations, such as the determination of thresholds for the penetration depth or the gradient based feature detection.

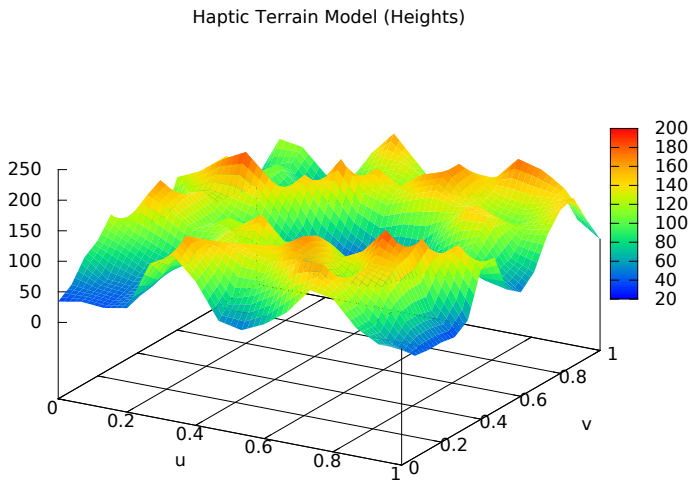


Figure 10: Visualization of the height values from the raster image

Coordinates

Along with the height the remaining components of the polar coordinates and the artesian coordinates will be stored for each rater point. The direct access of the coordinates will increase the speed by preventing the application from performing a reverse mapping to find the corresponding coordinate for a requested raster point.

Material

Inferences of the terrain materials can be obtained from the existing spectrum imaging or radar sounder analyses, but will not be considered. There will be no distinction between rocks, sand or any other material consistence and material composition in the haptic terrain model. It is expected that neither the profile liner, nor the volume measurer will benefit from such kind of information.

Stiffness

Stiffness is the most important characteristic in the haptic terrain model and determines the hardness of an object. A soft material such as foam has a low stiffness, the counterforce increases slowly when squeezing it. The terrain should not simulate a soft behavior, it should be hard to facilitate a precise positioning on the surface with high accuracy. The stiffness value will be global for the whole surface. There will be no local distinctions that needs to be stored in the haptic terrain model. The used stiffness value will be the maximum of the haptic device capabilities For the Phantom Omni the maximum stiffness is 1.02 $\frac{N}{mm}$. Preliminary tests have shown that the usage of

the maximum value leads to an abnormal behavior of the Phantom Omni near the boundary of the workspace range. The stiffness for the terrain model will be reduced to 0.75 $\frac{N}{mm}$ to avoid this issue. The maximum force that will be used for the terrain penetration is 3 N. Regarding the stiffness, this maximum force will be reached, when the penetration of the terrain is higher than 4 mm. The current workspace mapping returns the corresponding distance in model coordinates. This distance is stored within the haptic terrain model to speedup the calculation of the force depending on the penetration depth. It should also be noted that the squared value of the distance will be used instead of the euclidean distance value. The reason is the computation of the penetration depth, which will also be squared to save the additional square root operation.

Friction

Friction is a great feature to enhance the realistic feeling conveyed to the user, but it also requires additional calculations. It can be expressed as a force component that is parallel to the surface and opposes the motion direction. The magnitude depends on the pressure on the surface and a coefficient that is usually derived from the material. As mentioned above, information about the material is hardly available and not of interest. Regarding the objective of this work, there is no other attribute or property that can effectively be used as coefficient for friction. Also adding a default synthetic friction for the whole terrain has neither a supportive, nor a disadvantageous impact. As a result of this, friction will be ignored to save the additional force calculations and prevent exception handling for raising side effects such as circular opposing forces that leads to vibrations.

Roughness

Introducing roughness will also provide more realism. Roughness emerges when the density of available height samples is much higher than the samples which are represented on the output device. In a Geographic Information System (GIS) that is capable of visualizing a whole planet, this situation often occurs when the viewport is far above the ground and shows a level of detail which is different from the original level of detail. The samples between the reduced displayed resolution and the highest available resolution are used to calculate local roughness values from height variances. Such computation will become really complex and costly and will decrease the refresh rate of the haptic rendering algorithm. Besides the variance from height samples, there is no other attribute or property that can be profitable mapped to roughness regarding the purpose of this work. The presentation of roughness through a haptic device will usually be achieved by micro vibrations. Since accurate placement on the surface is one of the most important requirements, any kind of vibration is considered as a hindrance when trying to place a mark on the

terrain. To avoid all these disadvantages roughness will not be taken into account.

Gravitation and Magnetism

Gravity is the force that attracts a virtual mass of the user's tool towards the surface. There is no case where a tool benefits from gravitational influences in any way. The same applies to the magnetic field of a planet, which will also affect the user interaction. Both properties will be ignored for the haptic terrain model. However, attracting forces will be used in a different manner. A magnetic flux will be assigned to each point of the surface which may attract the user's tool. This is really useful when used in combination with virtual fixtures, which will be discussed later in this chapter.

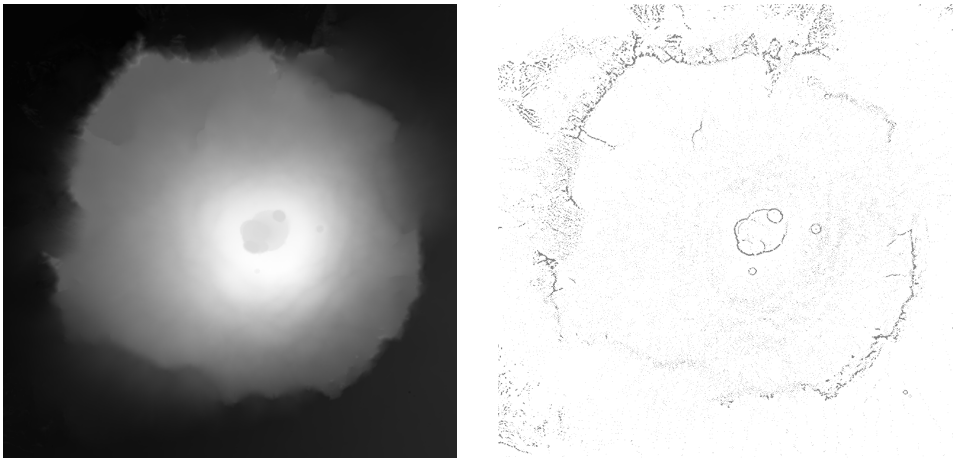
Flux Extraction for the Volume Measurer

In order to determine the flux values, it is necessary to evaluate the convex terrain features (required by the volume measurer). The detection of terrain features is not related to the context of this work, but unfortunately it is not implemented in the current version of the Virtual Reality terrain visualization and exploration framework (MarsVis). To ensure the functionality of the volume measurer the evaluation of terrain features is indispensable. Therefore a basic feature detection will be presented with the objective to provide some meaningful results that can be used for the flux determination. A simple approach is to extract features by applying a differential edge detection of second order to get the average slope differences. For this purpose the height values of the raster image will be convolved with a Laplacian kernel. For further performance optimizations, the version excluding diagonals will be chosen. The basic kernel will also be resized to the size of 7×7 to improve the detection of consistent features by skipping the two direct neighbors. An additional post-processing filter with a 3×3 blur kernel will be applied to reduce the remaining noise.

$$|E| = \begin{bmatrix} 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 4 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 \end{bmatrix} \qquad |B| = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \tag{7}$$

The resulting flux values will be normalized through division with the maximum resulting flux value. At the same time the normalized flux values are cutoff below a certain threshold. The threshold will remove negative values which are concave features and also remove values that are to smooth. The

chosen threshold depends strongly on the terrain structure of the current raster image, which makes this algorithm unreliable when changing the location. A flat terrain requires a different threshold than a rough terrain. Preliminary tests have shown that a threshold of 0.1 delivers acceptable flux maps, at least for the regions that will be used in the pilot study.



(a) Heightmap (b) Resulting fluxmap (inverted)

Figure 11: Feature detection and flux determination(Olympus Mons)

3.2.3 Conclusion

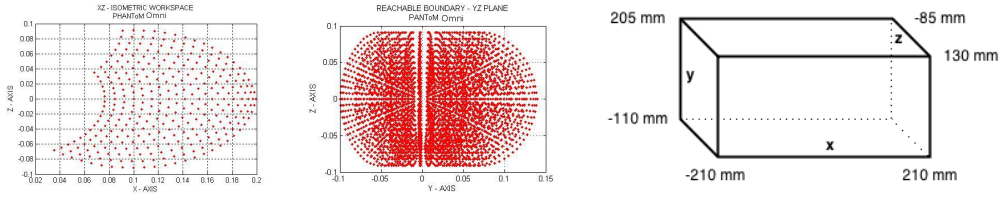
A raster image with a maximum resolution of 4096×4096 units will be used as haptic terrain model. Each point in the raster contains the value for the corresponding height, the magnetic flux and the related coordinates of the virtual space. The haptic terrain model presents only the current region that is visible and mapped to the workspace of the haptic device. When the viewport changes, the haptic model will be updated with the content of the new related visible region that is mapped to the workspace of the haptic device.

3.3 Workspace Mapping

To interact with the virtual model through the haptic device, a mapping of coordinates is required. The workspace mapping is responsible for various coordinate transformations between the coordinate systems of the workspaces from the haptic device and the different models. A fixed coordinate transformation to map the whole planet within the workspace of the haptic device will be inconvenient. Instead of such a static coordinate transformation, a dynamic mapping should be used, which only maps the portion of the planet to the haptic device workspace, which coincides with the currently visible area. Every time when the camera in the virtual model transforms, an update of the workspace mapping will be required.

3.3.1 Haptic Device Workspace

The haptic workspace is the area where the haptic device is operated. This space is limited by the hardware capabilities. The reachable maximum workspace of the used Phantom Omni haptic device is complex and hard to describe with functions.



(a) Side view (b) Front view (c) Bounding box

Figure 12: Phantom Omni reachable workspace (Source: [31])

Working with this complex workspace will be different for calculations such as intersection tests. It is more convenient to wrap this workspace in a bounding box. The reachable workspace is a strict subset of the defined bounding box. The dimensions of this box for the Phantom Omni haptic device are $420 \times 315 \times 215\ mm$ (x, y, z). The longest diameter within this box is $568\ mm$.

3.3.2 Virtual Model Workspace

The virtual model also referred to as simulation model is the representation of the planetary data. This model is already implemented in the MarsVis. The planet is centered in the origin of a Cartesian coordinate system and the poles are aligned to the z-axis (Figure 13). Simultaneously the coordinates are available with their latitude and longitude equivalents. A camera and the corresponding frustum are used to determine the viewport for the portion of the planet that should be rendered on the screen. The virtual model utilizes the trackball metaphor, where the camera's position and orientation is transformed instead of the planet. The same transformation can be used to convert coordinates between the local coordinate system of the camera and the global coordinate system of the planet.

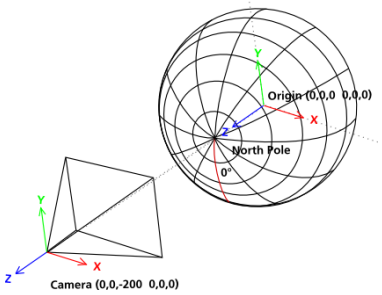


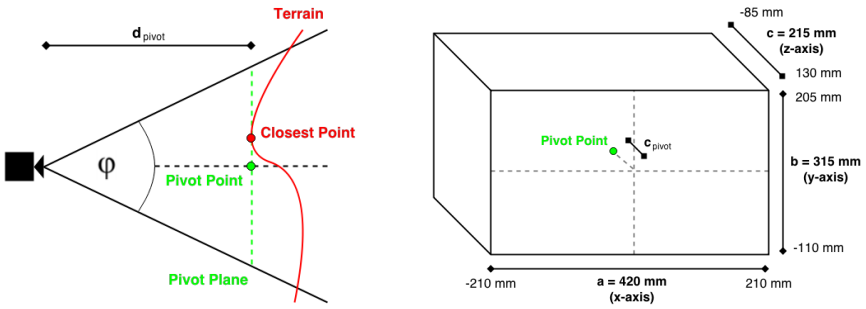
Figure 13: Workspace of the virtual model

3.3.3 Determine the Transformation

A region of the planet is limited between a latitude and longitude range which depends on the viewport of the camera. The basic idea is to translate and scale the workspace of the haptic device to fit in the camera frustum of the virtual model. The coordinate systems of both spaces are coincident. Empty space between the camera and the terrain should be eliminated. The transformation should ensure that the terrain close to the camera is reachable. The left, right, top and bottom boundaries of the camera frustum should be reachable even in the distant.

Pivot Point

The pivot point is some kind of reference point which forms the base of the transformation between the haptic device and the virtual workspace. Empty space between the camera and the terrain should be excluded. The mapped workspace of the haptic device should start slightly before the closest point of the terrain. The pivot in the virtual workspace is the center point of a clipping plane in the frustum (Figure 14a), which contains the closest terrain point to the camera. The value d_{pivot} is the distance (positive) between the camera and the pivot point. The corresponding pivot in the workspace of the haptic device (Figure 14b) is also centered in x, y direction. To prevent a mapping of the closest terrain point to the reachable boundary of the haptic device, the pivot will be shifted along the z-axis away from the boundary, into a more effective region. For the Phantom Omni haptic device, a z-shift of $c_{pivot} = 100\ mm$ will be used for the pivot point. The z-coordinate of the pivot point is $c_z = 30\ mm$.



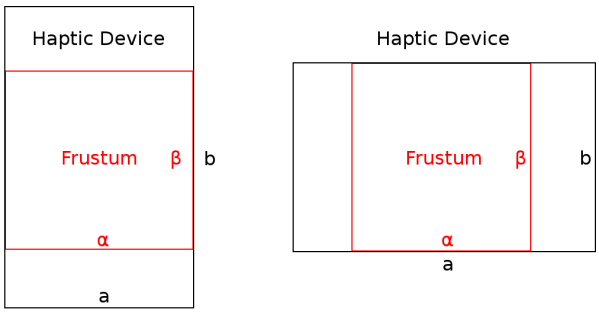
(a) Virtual workspace (b) Device workspace

Figure 14: Pivot point

Workspace Scaling

After the detection of the pivot point, the workspace of the haptic device needs to be scaled. The scaling of the haptic device workspace is based on the z-axis. The other dimensions will be fitted with respect to keep the original ratio. The behavior of the scaling operation should be the same like the "zoom to fit"

in image processing. The box of the haptic device workspace will be scaled along the z-axis until it fully includes the camera frustum. The scale factor should be as small as possible to fulfill this requirement. Let a, b and c be the width, height and depth of the haptic device workspace. From the previous section c_1 is applied as the z-offset of the pivot point. The angles α and β define the frustum of the camera in x-direction and y-direction. Depending on the difference between the $\frac{a}{b}$ ratio of the haptic device workspace and the $\frac{\alpha}{\beta}$ ratio of the frustum, the value of the corresponding dimension will be chosen to determine the scale factor, see Figure 15.



(a) $\frac{a}{b} < \frac{\alpha}{\beta}$ (b) $\frac{a}{b} \geq \frac{\alpha}{\beta}$

Figure 15: Ratio differences between haptic workspace and frustum

The synonyms $s \in [a, b]$ and $\varphi \in [\alpha, \beta]$ will be used for the side and angle that corresponds to the met condition. The values for s and φ are assigned depending on the following comparisons.

$$s = \begin{cases} a & \frac{a}{b} < \frac{\alpha}{\beta} \\ b & \frac{a}{b} \geq \frac{\alpha}{\beta} \end{cases} \qquad \varphi = \begin{cases} \alpha & \frac{a}{b} < \frac{\alpha}{\beta} \\ \beta & \frac{a}{b} \geq \frac{\alpha}{\beta} \end{cases} \tag{8}$$

The next step is to determine the scale factor n along the z-axis for the selected dimension.

$$n = \frac{d}{c - c_{pivot}} \tag{9 .1}$$

$$d = \frac{d_{pivot}}{\left(\frac{s}{2(c-c_{pivot}) \cdot \tan\left(\frac{\varphi}{2}\right)} - 1 \right)} \tag{9 .2}$$

$$n = \frac{d_{pivot}}{\left(\frac{s}{2 \cdot \tan\left(\frac{\varphi}{2}\right)} - (c - c_{pivot}) \right)} \tag{9 .3}$$

The derivation of the formula is based on trigonometric relations shown in figure 16.

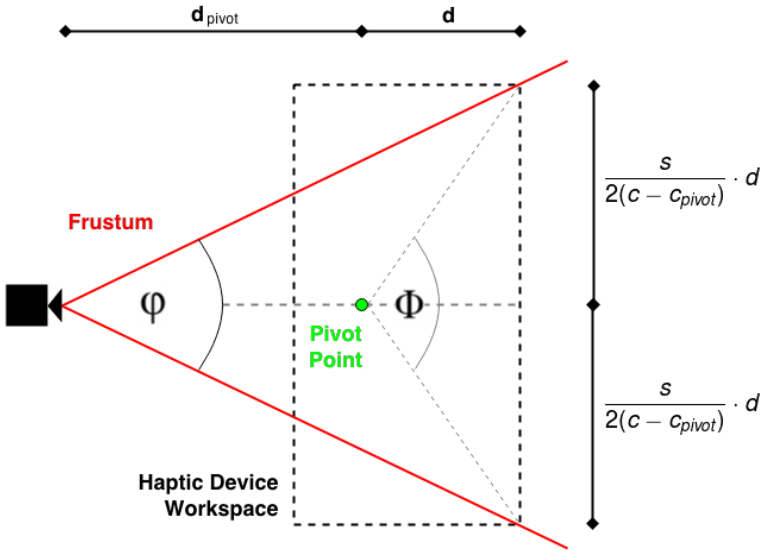


Figure 16: Scale the workspace

There is a possibility that the box of the haptic device workspace will not reach the frustum boundaries through scaling. This happens when the angle φ of the frustum is equal or larger than the angle ϕ. To ensure that this will not happen, φ must fulfill the following requirement.

$$\frac{s}{c - c_{pivot}} > 2 \cdot \tan\left(\frac{\varphi}{2}\right) \tag{10}$$

Mapping from Haptic Device to Virtual Model

After the pivot point and the scaling factor have been computed, a transformation matrix can be composed. The value d_z is the z-coordinate of the pivot point in the virtual workspace and c_z is the z-coordinate of the pivot point in the haptic device workspace. The difference between the scaled c_z and d_z is used to determine the translation for the z-coordinate. The coordinate systems of the Phantom Omni haptic device and the virtual camera are coincident. As a result of this, no additional transformations, such as inverting values, are required.

$$|T_1| = \begin{bmatrix} n & 0 & 0 & 0 \\ 0 & n & 0 & 0 \\ 0 & 0 & n & d_z - n \cdot c_z \end{bmatrix} \tag{11}$$

With $|T_1|$ the coordinates of a point from the haptic device workspace can be transformed to the camera coordinate system. The existing transformation matrix $|T_2|$ from the virtual model can be used to map the coordinates from the camera coordinate system to the global coordinate system of the planet. As already mentioned, the Cartesian coordinates are paired with their latitude and longitude values. Applying $|T_1|$ and $|T_2|$ to an arbitrary point in the haptic device workspace, returns the corresponding Cartesian and Polar coordinates of the virtual model.

Mapping from Haptic Device to Haptic Model

This section will deal with the coordinate mapping from the workspace of the haptic device to the workspace of the haptic model. With the previous transformation it is already possible to convert the coordinates to the virtual workspace. The missing component in the chain is the conversion between the virtual workspace and the haptic model workspace. The longitude and latitude of the virtual model can be linearly mapped to the u, v coordinates of the haptic model. Let lg_{min} and lg_{max} be the longitude interval and lt_{min} and lt_{max} be the latitude interval of the selected region. The following linear mapping functions will be used to compute the corresponding u, v coordinates. A special case occurs, when the zero median lies within the selected region and the minimum longitude is larger than the maximum longitude. The maximum longitude has to be incremented by an additional rotation of 2π, which will increase the value, but maintain the position of the coordinates. The input latitude needs to be changed as well. When the maximum longitude is larger than 2π and the input latitude is smaller or equals the original maximum longitude (before modification), then the latitude will also be supplemented by 2π.

$$v(lt) = \begin{cases} n.d. & lt_{min} = lt_{max} \\ \frac{1}{lt_{max} - lt_{min}} (lt - lt_{min}) & lt_{min} < lt_{max} \end{cases} \tag{12}$$

$$u(lg) = \begin{cases} n.d. & lg_{min} = lg_{max} \\ \frac{1}{lg_{max} - lg_{min}} (lg - lg_{min}) & lg_{min} < lg_{max} \\ \frac{1}{(lg_{max} + 2\pi) - lg_{min}} (lg - lg_{min}) & lg_{min} > lg_{max}, lg \geq lg_{min} \\ \frac{1}{(lg_{max} + 2\pi) - lg_{min}} ((lg + 2\pi) - lg_{min}) & lg_{min} > lg_{max}, lg < lg_{min} \end{cases} \tag{13}$$

The formulas require the latitude and longitude interval of the selected region. The first step to determine these boundaries is to transform the full workspace of the haptic device to the virtual model. This can be done by just mapping the eight corner points. The selected region is the intersection between the transformed workspace of the haptic device and the planet in the virtual space. There are plenty of possibilities such as testing the intersecting line segments of the box, testing the intersecting side planes of the box, splitting the box into simplexes for barycentric representation or wrap the box into another bounding sphere or axis aligned bounding box and perform the intersection test with those. The implementation will use existing open source code to compute the intersection points between a sphere and a box. There are two special cases that needs to be noted. The box may totally include the sphere thus there is no intersection. In this case, the whole sphere will be mapped to the haptic terrain model. It may also occur that multiple independent regions emerge through the intersection. This happens when the diameter of the box is slightly larger then the radius of the sphere. In such a case, where the diameter

of the box is larger than the radius of the sphere and at least two unconnected regions emerge, the whole sphere will be mapped to the haptic terrain model.

3.4 Fixtures for Geodesy Tools

The following sections will discuss the requirements and design for virtual fixtures that will improve the selected geodesy analysis tools.

3.4.1 Profile Liner

The first tool that will be investigated is the profile liner. A profile line consists of a start point and an endpoint. All height values along this line will be sampled and plotted in a diagram. For planetary researchers the evaluation of a single profile line is not sufficient to find useful geoscientific conclusions. Multiple profile lines which are plotted in the same diagram are more informative, but an arbitrary orientation of the lines is inconvenient for comparison and analysis tasks. The following constraints briefly explain the desired alignment of multiple profile lines.

Start Point Constraint

Each start point of the following lines should be aligned along a certain plane. This plane needs to be composed only once, after the start point of the first line has been set. The plane contains the start point of the first line (s_1) and is described by two vectors, that spans the plane. The first vector is formed between the start point of the first line and the origin of the planet. The second vector in the plane is the cross product of the first vector and the vector between the start point (s_1) and the endpoint (e_1) of the first line. The plane containing the start point of the first line will be referred to as start point plane (SPP).

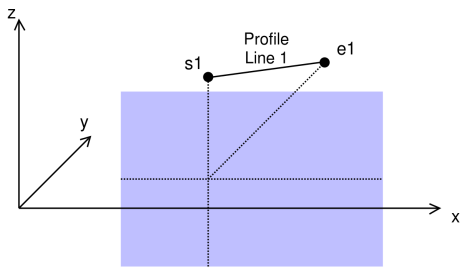


Figure 17: SPP for placing start points of sequential lines

Endpoint Constraint

After the start point of a following line has been set, the corresponding endpoint needs to be set as well. The endpoint should also be aligned along two

particular planes. The conditions for the first plane are the same as described for the SPP, except that the endpoint of the first plane (e_1) will now be used as reference. Lines that are oblique regarding the orientation of the first line will distort the profile and may lead to wrong assumptions. The profile lines should preferably be parallel to the first line to be consistent. The meaning of parallel in the context of spherical coordinates is referring to the latitude and longitude distance. In addition to the first plane, a second plane, that is perpendicular to the first plane and perpendicular to the terrain, will be used . This plane contains the start point of the current line (s_2) and is spanned by two vectors. The first vector is formed between the start point of the current line (s_2) and the origin of the planet. The second vector is the normal vector of the first plane. The plane containing the endpoint of the first line will be referred to as endpoint plane (EPP). The plane containing the start point of the current line and perpendicular to the EPP will be referred to as equidistant plane (EQP).

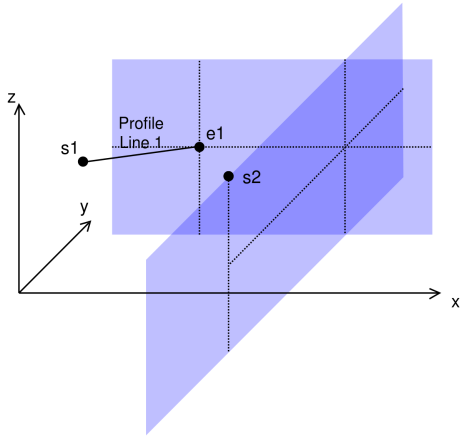


Figure 18: EPP and EQP for placing endpoints of sequential lines

Fixture

The basic procedure to draw a set of profile lines with a virtual fixture can be split into various different steps. The first step is to freely place the start point and the endpoint of the reference profile line. A virtual fixture will be used to assist in placing the start point of the next line within the SPP. After placing the start point of the next line, the virtual fixture will be updated to place the corresponding endpoint. The virtual fixture is updated to assist placing the endpoint within the EPP and EQP. After placing the endpoint of the current line, these steps can be repeated until the set of profile lines is completed. Most of the techniques for virtual fixtures are inconvenient for usage with the profile liner. Repelling is a fixture where the user is forbidden to enter a certain area. The user will be pushed from that prohibited area to a permitted area. The only permitted area is within the planes. This limits the user's freedom

drastically when moving outside of the planes. A penalty-based vibration will also distort the user when outside of the planes. It is difficult to place a point accurately on the terrain when the interface is vibrating. Snapping the user to the planes when within a certain range is considered to be the most efficient fixture. It will support the user to set the start and endpoints in the correct planes, without interfering when moving in an area outside of the planes. The directional attracting force within the snapping range allows a better accuracy when placing points on the terrain than a penalty based vibration fixture. The calculation of the attracting force will be discussed in section 3.4.3 and the influence on the resulting force in section 3.5.8.

3.4.2 Volume Measurer

The volume measurer uses a poly-line which consists of a list of control points defined by the user. In most cases the points should be set along certain terrain features such as crater rims, food of mountains or canyons. From the structure of a terrain two classes of features can be derived. When the slope over a given surface sample decreases than it must be a mountain or rim which has a convex characteristic on the planetary surface. In that case the difference of the slopes is negative and a higher absolute value means a steeper edge. On the other hand when the slope increases than a valley or ravine can be expected which is concave regarding the planetary surface. For a concave feature the difference of slopes is positive and the absolute value reflects the incline. Considering a realistic based haptic simulation of the terrain, it seems easy to detect concave terrain features by applying pressure and slide into them.

Fixture

The detection of concave features is already covered by the realistic terrain simulation using the haptic terrain model and does not require an additional virtual fixture. The user can simply press the effector of the haptic device with a certain direction onto the virtual terrain and will automatically slide into the adjacent concave terrain feature. The main problem arises when the user wants to detect convex terrain features. When trying to place a pen tip on a sharp edge, the pen slips aside. The same will happen on a haptic supported simulation, where a virtual pen tip will also slip aside when placed on an edge in the virtual model. This will make it difficult for users to place the control points on convex features. A virtual fixture should assist to find this class of features and compensate this issue. The basic idea is to attract the user to terrain features when setting the points of the poly-line. The haptic terrain model was designed to support the storage of magnetic flux values to realize such kind of attraction.

3.4.3 Snapping

Both force feedback supported geodesy analysis tools utilize snapping as virtual fixture. Snapping will attract the user from the current location to the snapping point. When the attracting object is a line or a plane, the snapping point is the closest point within the line or plane regarding the current location. The direction vector emerges from the current location and the snapping point. The magnitude of the force depends on the distance, the closer the current location is to the snapping point, the higher is the magnitude of the attracting force. Haptic devices have a discrete refresh rate to update the force. The high attraction force at a distance close to zero will result in a push through the snapping point. During the time period of an update cycle a distance emerged on the opposite direction. When the device updates the force again, another push through the snapping point is initiated. The device get caught in an oscillating loop which can be imagined as a ping-pong effect. A solution is to damp the attraction force when the distance is smaller than a minimum (d_{min}). The force at the distance of d_{min} is limited to a maximum (F_{max}), which is convenient for the user, or is limited by the haptic device capabilities. Another constraint is the scope of the snapping point, that should only be valid in an interval between a maximum distance (d_{max}). At the distance of d_{max} the force should be zero as well. A general approach is to use a linear function based on Hooke's law. Another opportunity based on Newton's law of universal gravitation provides a more natural non-linear behavior of the calculated force. A hyperbola function offers a more non-linear behavior and keeps the complexity of the function below Newton's law of universal gravitation. In a preliminary test a hyperbola function and a linear approach were compared. The low fidelity of the device and the lack of human perception to notice small force differences, made it impossible to distinguish between both functions. The linear approximation is sufficient and will be realized with the formula presented in Equation 14.

$$F(d) = \begin{cases} 0 & d < -d_{max} \\ \frac{F_{max}}{d_{max} - d_{min}} (d + d_{max}) & -d_{max} \leq d \leq -d_{min} \\ -\frac{F_{max}}{d_{min}} d & -d_{min} > d > d_{min} \\ \frac{F_{max}}{d_{max} - d_{min}} (d - d_{max}) & d_{min} \leq d \leq d_{max} \\ 0 & d > d_{max} \end{cases} \tag{14}$$

The equation considers negative values which indicates the direction of the distance or force along the given vector. Figure 19 shows an example graph of the force within the snapping distance d_{max} for both, the coinciding and non-coinciding direction.

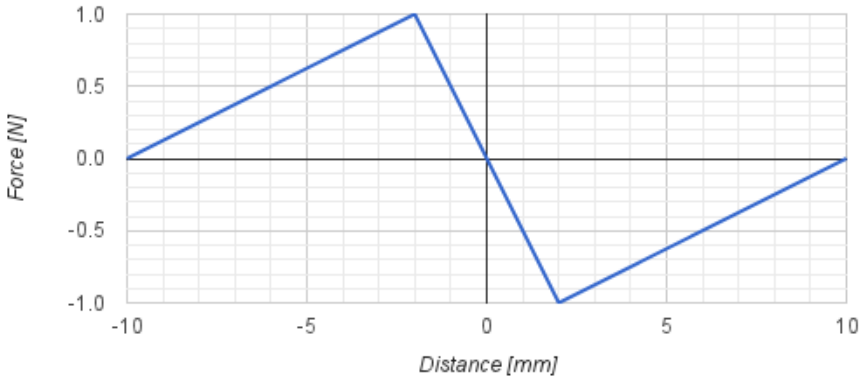


Figure 19: Linear snapping with $F_{max} = 1, d_{max} = 10, d_{min} = 2$

3.5 Haptic Rendering

Haptic rendering is a collection of connected algorithms that interact with the simulation model and the haptic device. The input location from the haptic device will be acquired and processed within the simulation to compute a corresponding output force, that is returned to the haptic device. The haptic rendering will be based on the fundamental pipeline concepts, introduced in [16]. Furthermore, the simulation model which consists of the DTM, will be extended by the additionally introduced haptic model and workspace mapping, see Figure 20.

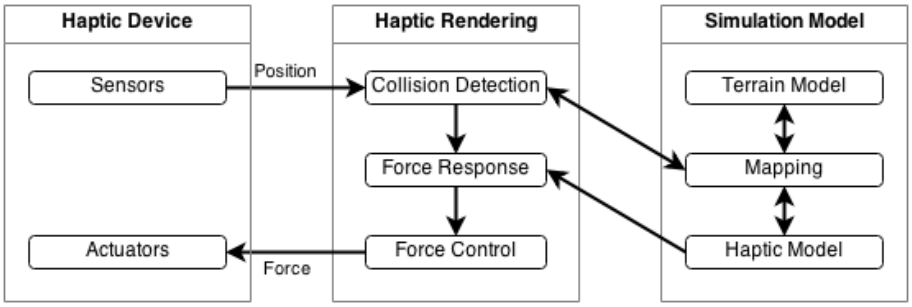


Figure 20: Components for haptic rendering

The haptic interaction will be limited between the terrain, the avatar and the virtual fixture for the corresponding tool. The appearance and behavior of the avatar is influenced by the used tool. While the avatar can be shaped differently, the haptic interaction is limited to a single point of the avatar.

The force response block can be complex with a set of sophisticated algorithms depending on the constraints of the simulation model. Regarding the given constraints for terrain rendering and virtual fixtures, the force calculation will be split into a chain of three sub-blocks. First the resulting force emerged from

the avatar collision will be calculated. Afterwards this force will be shaded, before it will be combined with the force calculated from the virtual fixture. In Figure 21 the steps are presented as flowchart.

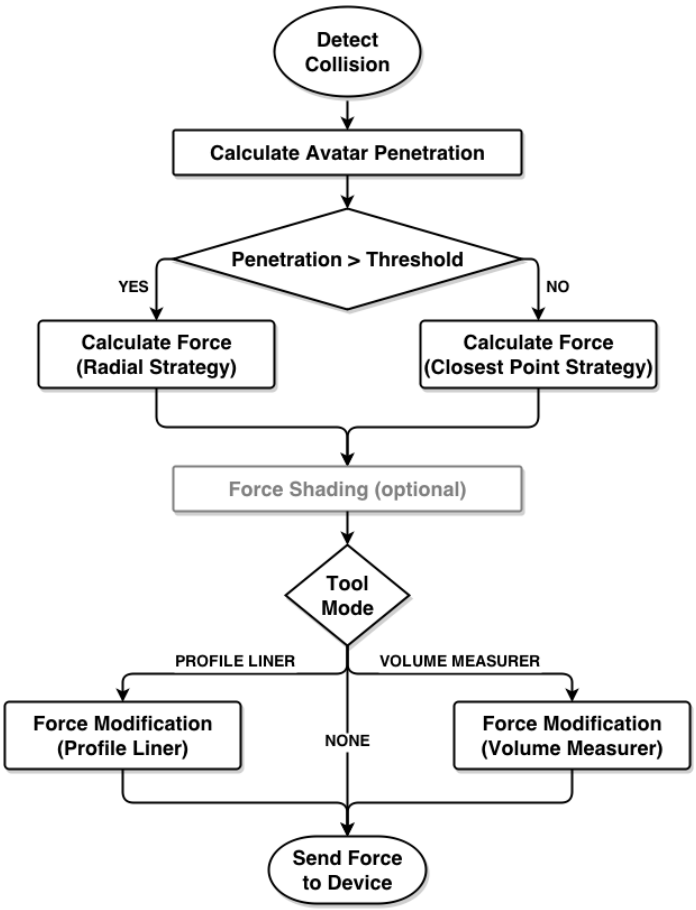


Figure 21: Detailed haptic rendering pipeline

3.5.1 Collision Detection

To decide if the avatar collides with the terrain, an inside check has to be performed. The current position of the avatar is related to the position of the haptic device's end effector. After transforming the position to the virtual workspace, the distance between the origin of the planet and the avatar can be calculated. Simultaneously the latitude and longitude coordinates of the avatar will be available. Those can be used to acquire the corresponding height value of the terrain from the DTM. A collision occurs, when the avatar lies below the terrain. That is the case, when the distance from the avatar to the origin is less or equal than the height value of the terrain.

3.5.2 Penetration Force

After a collision between the avatar and the terrain has been confirmed, a resulting force needs to be calculated. The objective of this force is to push the avatar above the terrain. Two strategies will be used to determine the direction and the magnitude of the responding force, the *closest point strategy* and the *radial direction strategy*.

3.5.3 Closest Point Strategy

An avatar that penetrates the terrain should be pushed to the next terrain point that is closest to the avatar. Let $(x_2-x_1)^2+(y_2-y_1)^2+(z_2-z_1)^2$ be the metric to measure the (squared) distance between two points. This strategy starts with the assumption, that the closest point is the terrain point directly above the avatar.

Procedure

The distance between the avatar and the above terrain point is the initial penetration depth. This value has already been calculated to decide which strategy should be used. In the next steps all distances between the avatar and the surrounding terrain points (Figure 22) will be calculated and compared to the initial penetration depth.

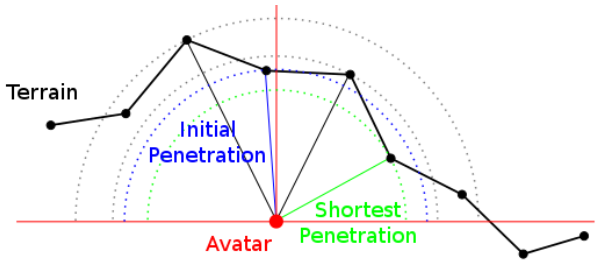


Figure 22: Distances between avatar and surrounding terrain points

This method takes advantage of the raster alignment in the haptic terrain model. The surrounding terrain points are scanned in order along an expanding rectangle, starting with the initial u, v location of the avatar, see Figure 23.

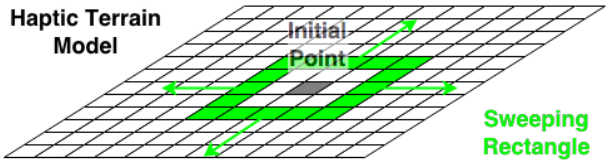


Figure 23: Sweeping rectangle

First the distances to the direct neighbors of the initial terrain point will be calculated. After that, the sweeping rectangle will be expanded to process the

terrain points next to the neighbors and so on. When one of the calculated distances is smaller than the initial penetration depth, this distance will replace the value of the initial penetration depth. At the same time the direction vector between the avatar and the terrain point will be saved as well. Assuming a terrain point to be the closest point of the terrain is an approximation. The real closest point lies within one of the four adjacent triangles. Figuring out which triangle contains the closest point, determine the barycentric coordinates of this point within the triangle and finally compute the resulting force direction and penetration depth, is really expensive and will not be applied. The accuracy will not benefit from this triangle interpolation, because the haptic terrain model was designed to support a resolution close to the physical resolution of the haptic device.

Abort Condition

This strategy is inefficient when all raster points of the haptic terrain model needs to be checked. After the sweeping rectangle reaches a certain size, it is impossible that any terrain point outside of the rectangle are closer to the avatar than the current point inside the rectangle. This certain size is limited by the distance to the current closest point. When the smallest distance between the rectangle and the avatar is larger than the current distance between the avatar and the closest terrain point, the search can be stopped. The search into a direction will also be stopped, when the corresponding side of the sweeping rectangle reaches the boundary of the haptic terrain model.

Updating the Sweeping Rectangle Boundary

It is necessary to determine the amount of raster points in the u, v direction which should be processed by the sweeping rectangle. The length of the arcs covered by the number of raster points must at least equal the distance to the closest point. The raster representation of the haptic terrain model is linear for the v component. The arc length that is spanned between $v = 0$ and $v = 1$ is stored in the haptic model and can be used to calculate the minimum and maximum v values and the corresponding number of raster points in v-direction for the sweeping rectangle depending on the current penetration depth. The u component is non-linear, the covered arc length between $u = 0$ and $u = 1$ differs depending on the corresponding v component. The arc length that is spanned between $u = 0$ and $u = 1$ is stored for each v in the haptic model. For all arcs along the u direction between a v-interval, the minimum arc must be at one of the v-interval boundaries. To determine the minimum and maximum u values for the sweeping rectangle, the arcs at the minimum and maximum v are compared. The smaller arc will be used to calculate the number of raster points that span the sweeping rectangle in u direction.

Limitations

An exception occurs when one or both poles are included in the haptic terrain model and lies within the v interval of the sweeping rectangle. The arc length at a pole is zero. As a result the sweeping rectangle will be expanded over the whole haptic terrain model. It can be assumed that the high cost will reduce the update rate drastically. Direct haptic interaction with the region closely to the poles will be excluded from the evaluation.

Processing Time

It has already been mentioned that the cost depends on the penetration depth. The sweeping rectangle grows from the center of the initial penetration location. An iteration is the inclusion of the surrounding raster points of the current rectangle. For each raster point an operation consisting of vector composition, distance computation and distance comparison needs to be performed. The cost can be described with a big O notation of quadratic degree. The limit n depends on the number of raster points in the u, v interval which is limited by the penetration depth or the boundary of the haptic terrain model. The final composition of the responding force vector as described in section 3.5.5 with a constant cost, makes up the smallest part and is not mentioned within the equation.

$$O(n) = 1 + \sum_{i=0}^{n-1} 8i \tag{15 .1}$$

$$= (2n-1)^2 \tag{15 .2}$$

An impact on the refresh rate of the haptic rendering loop will be unavoidable for a larger n, even on an advanced hardware platform. The reduced refresh rate is still expected to be sufficient to perform the evaluation of benefits from the haptic supported geodesy tools.

3.5.4 Radial Direction Strategy

This method is much faster and simpler than the previous presented strategy. Instead of searching the direction to the closest terrain point, the orientation is opposing the radial segment between the avatar and the origin of the planet as shown in Figure 24. The direction vector can be simply calculated as the difference between those two points. Subtracting the distance between the avatar and the origin from the terrain height of the corresponding point above the avatar provides the penetration depth. This is the same penetration depth used to decide which strategy should be used at the beginning and also the same as the initial penetration depth from the previous presented strategy of the closest point.

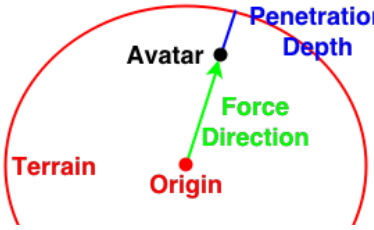


Figure 24: Radial force for high penetration depth

Limitations

This strategy is reliable at any position within the haptic terrain model. The non-linear mapping behavior of the haptic terrain model has no impact on the results of the direction or penetration calculation. The drawback of the inaccuracy was already explained and is the reason why this strategy will only be used when the penetration depth is relative high compared to the height deviation in the haptic terrain model.

Processing Time

There are only two values that need to be computed, the penetration depth and the radial direction vector. The processing time for the additional computation of the resulting force vector described in the following section 3.5.5 is also considered. The sum of the cost for these operations is constant. This computation should be fast enough on current hardware to support the highest possible refresh rate of the haptic rendering loop.

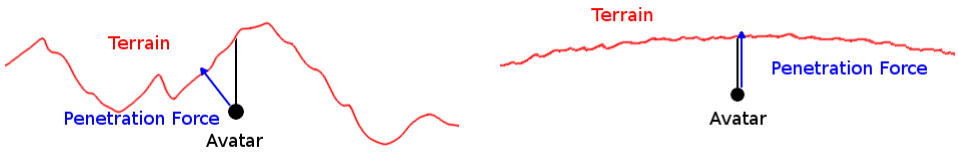
3.5.5 Force Magnitude and Vector

Both strategies will compute the direction of the force vector and the penetration depth along this vector. The strategies will be expanded by the calculation of the force vector. The force is calculated based on Hooke's law using the stiffness and the displacement. The haptic terrain model already holds the displacement (d) in model coordinates, where the maximum force (F_{max}) of 3 N will be reached. A linear mapping can be used to calculate the corresponding force magnitude (F) for the current penetration depth (s). When the penetration depth is higher than the displacement of the maximum force, it will be cut off, so the maximum force value will be used instead. Equation 16 shows that the resulting force vector is the product of the force magnitude and the normalized direction vector.

$$F(s) = \begin{cases} \frac{s}{d} F_{max} & s \leq d \\ F_{max} & s > d \end{cases} \tag{16}$$

3.5.6 Strategy Selection

To decide which strategy will be used, the penetration depth of the avatar needs to be calculated. The collision detection test already offers the height of the terrain and the distance from the avatar to the origin of the planet. The difference of both values is the distance of the avatar below the terrain at the current position. When the penetration depth is relatively small (Figure 25a), the force will be computed by the closest point strategy. This will deliver the most realistic feeling of discontinuities on the rigid terrain. This is an accurate but costly method and the computation power increases depending on the penetration depth. When the penetration depth is relatively high (Figure 25b), the direction change for the force vector is so slightly when applying the first strategy, that it will not be noticeable by the user anyway. So the idea behind the second strategy is to use the direction from the origin of the planet to the avatar. This will be much faster and a really good approximation.



(a) Small penetration (b) High penetration

Figure 25: Penetration depth relative to local heights

The selection of the applied strategy will depend on the current selected region that is represented in the haptic model and the penetration depth. The haptic model contains information about the maximum, the minimum and the average height. The threshold to decide which of the strategy will be used is based on the height difference of the minimum and maximum height compared to the penetration depth. The second strategy will only be used, when the penetration depth is below the average height minus the maximum height difference of the haptic terrain model. This will keep the derivation between the approximated direction and the direction to the closest terrain point relative low. It also prevents the usage of the costly first strategy where the deep penetration renders the accuracy redundant anyway.

3.5.7 Force Shading

To reduce the feeling of discontinuities a proper force shading algorithm can be applied to smooth the corners and edges of the terrain. This can be done by interpolating over the adjacent discrete values. The first strategy already uses an approximation to calculate the force, instead of interpolating the closest point to the terrain, it directly points to the closest discrete sample. Force shading will not work correctly with these approximated vectors and can not

be used anyway. Furthermore there are other reasons why force shading is not desired. The most important is the the processing time, force shading will increase the cost of the computation significantly. Furthermore, the force shading seems not necessary. It has already been mentioned, that the re-sampled haptic terrain model has a resolution close to the physical resolution of the haptic device. When the source (DTM) for the re-sampled data has a lower resolution than the sample rate of the haptic terrain model, the height values are interpolated by weighted sums of the surrounding values. With this prerequisite, the sub-accuracy of the haptic device can only be slightly higher than the discretization of the samples in the haptic terrain model. Another point of interest is the effectiveness. The haptic rendering will not benefit from force shading in the context of the research objective. Force shading will smooth the terrain features such as corners and edges. Regarding the geodesy tools, the priority is to detect terrain features. Using a force shading algorithm is counter productive and make it more difficult to feel those features. Currently the force vectors will push the user to the discrete samples when penetrating the terrain. Assuming the terrain features coincides with the discrete samples, this is a favorable side effect. Applying force shading will alter the force vectors and eliminate this positive side effect.

3.5.8 Virtual Fixtures

At this point the haptic rendering provides the force vector that works against the user's terrain penetration. This is sufficient to convey the realistic feeling of discontinuities on the surface. The missing components are the virtual fixtures. The force needs to be combined with the force from the virtual fixture of the currently activated geodesy to tool. There are three modes that decide which virtual fixture will be applied. The first mode is the navigation mode. This mode is barely to change the viewport of the camera and feel the terrain surface. No additional force calculation is required, the force can remain as it is now. The second mode is the profile liner mode where the user can draw the profile lines. The current penetration force needs to be combined with the snapping force. The last mode is the volume measurer mode. When setting points for the poly-line, the user is attracted to terrain features. The snapping force will also influence the current penetration force.

3.5.9 Profile Liner Mode

The workflow of the profile liner can be split into three phases.

Phase I

In the first phase, the user sets the start point and endpoint for the first line, which will be used as reference. During this phase, the current penetration force stays untouched. The first phase does not involve any snapping forces.

Phase II

The second phase is initiated every time when the start point of a sequential profile line is going to be set. The creation of the SPP has already been discussed and provides a point on the plane (s), two linear independent vectors spanning the plane ($\vec{v_1}, \vec{v_2}$) and the perpendicular unit vector ($\hat{n}$). The current location of the user (p) has a distance (d) to the plane. The equation of the plane in parametric notation is expressed in Equation 17 .1, where x is any point on the plane.

$$0 = n_x x_x + n_y x_y + n_z x_z + k \tag{17 .1}$$

The constant k can be determined (Equation 17 .2) by plugging in the values for the given point s into the Equation 17 .1.

$$k = -n_x s_x - n_y s_y - n_z s_z \tag{17 .2}$$

The general rule to calculate the distance between a point and a plane in parametric form will be used. The normalization was canceled out because the Equation 17 .3 is already based on a unit vector.

$$d = n_x p_x + n_y p_y + n_z p_z + k \tag{17 .3}$$

The resulting distance will be signed. The force magnitude (F) can be calculated from the absolute value of the distance. Furthermore the sign is important to determine the correct direction of the snapping force vector. If the distance is negative, the direction of the snapping force ($\hat{f}$) is co-directional with the normal unit vector ($\hat{n}$), otherwise it is opposing the normal unit vector ($-\hat{n}$). The final force vector ($\vec{f}$) can be retrieved by multiplying this snapping direction vector ($\hat{f}$) with the force magnitude (F).

Phase III

The third phase is initiated after a start point of a sequential profile line was set and the endpoint needs to be set. The scheme is the same as described in the second phase, except that other snapping planes will be used. Instead of the SPP, the EPP and EQP are used to calculate the snapping force. This results in two snapping force vectors ($\vec{f_1}, \vec{f_2}$), one for each plane. Both forces are perpendicular, because the snapping planes EPP and EQP are perpendicular. The final snapping force ($\vec{f}$) is the sum of the snapping vectors ($\vec{f_1}, \vec{f_2}$). With a maximum magnitude of F_{max} for the component vectors ($\vec{f_1}, \vec{f_2}$), the maximum magnitude of the final snapping force ($\vec{f}$) increases to $\sqrt{2} \cdot F_{max}$. When the magnitude of this force vector ($|\vec{f}|$) exceeds the maximum snapping force magnitude (F_{max}), the vector will be scaled by the factor $\frac{1}{\sqrt{2}}$.

3.5.10 Volume Measurer Mode

There are multiple snapping points (terrain features) within the snapping distance (d_{max}), but only the point with the highest snapping force is allowed to attract the avatar. To check the snapping force for all terrain points, the presented rectangular sweep algorithm from section 3.5.3 is adapted. There are some significant changes to the procedure that needs to be mentioned. The direction and distance to the point with the highest flux/distance ratio is stored, instead of the distance and direction to the closest point. The interval of the sweeping rectangle equals the defined snapping distance (d_{max}) instead of the distance to the closest point. The snapping force magnitude for each point is calculated as described in section 3.4.3 depending on the corresponding distance. Furthermore the snapping force magnitude is multiplied with the corresponding flux value (interval $[0, 1]$) of the terrain point.

3.5.11 Combining Forces

The penetration force is calculated using the radial strategy, which has the side effect of sliding upwards at a slope. After calculating the penetration force and the snapping force, both are combined by addition. The penetration force magnitude is within the interval $[0, 3]$, while the snapping force magnitude lies between $[0, 1]$. The sum of both vectors results in a force vector with a magnitude ranged between 0 and 4. The Phantom Omni haptic device is limited to 3.3 N, so the final force vector needs to be scaled with the factor $\frac{3.3}{|\vec{f}|}$, when the maximum force magnitude of the device is exceeded.

3.6 Conclusion

This chapter reviewed a selection of commercial haptic devices. The available Phantom Omni haptic device was rated as appropriate for haptic supported terrain exploration and analysis and can be used without concerns. An additional raster based haptic terrain model was introduced, which provides the terrain and flux data of the currently visible/reachable region for the haptic rendering pipeline. A workspace mapping with ideal fitting to the visible/reachable region aids in the coordinate transformation between the workspaces of the DTM, the haptic terrain model and the haptic device. A haptic rendering pipeline connected all related components to calculate a response force for terrain penetration. The core of the rendering pipeline consists of two different vector field strategies to compute the response force depending on the penetration depth. While the slow and precise strategy is used for slight penetrations, the fast and less accurate strategy is used for deep penetrations. Finally the haptic rendering pipeline was complemented by a virtual fixture for the profile liner that snaps the user to a grid-like pattern while drawing lines on the terrain and by a virtual fixture for the volume measure that attracts the user to rims (e.g. mountain ridges).

4 Implementation

After discussing the concepts and design, this chapter will outline the implementation for the haptic supported geodesy tools. The implementation will produce a non optimized prototype, which meets the requirements to evaluate the haptic supported workflow and to assess the benefit.

4.1 Used Software

The objective of this work was to extend the existing Virtual Reality terrain visualization and exploration framework (MarsVis) with haptic support. Naturally it uses the existing sources sch as MarsVis itself and the underlying ViSTA framework. Additionally the OpenHaptics Application Programming Interface (API) will be added for advanced communication with the haptic device.

4.1.1 MarsVis

The MarsVis application consists of two modules. The core module is a shared library written in C++ using boost to expose some functionalities to python. Essential functionalities such as the processing of the Digital Terrain Model (DTM), scene organization, graphic rendering and various Virtual Reality (VR) are embedded in this library. The most important dependency of this library is the ViSTA framework which will be introduced in the following section 4.1.2. The second module is a set of python scripts built on top of the C++ library. The boost libraries are used to expose the selected C++ functions to python. The scripts are responsible to start the application, instantiate objects and control the interaction with the functionalities of the shared library.

4.1.2 ViSTA Framework

ViSTA is an extensive C++ framework do develop interactive VR applications. Some of the main features are the scene graph management, the dataflow network, the interface to various (VR) devices, the distributed clustering and the display management and stereoscopic rendering. ViSTA is the main engine of the MarsVis library. It also offers various system independent tools such as threading, file I/O, timers, vector and matrix math, event observers and much more.

4.1.3 Boost

Boost is a set of C++ libraries to increase the productivity. The functionality ranges from linear algebra calculation to unit testing. The relevant part of boost that will be used, is the interface library to enable interaction between python and C++.

4.1.4 OpenHaptics

The OpenHaptics API is a set of C++ libraries, allowing the interaction with supported Geomagic (Sensable) haptic devices such as the Phantom Omni. The API is split into two different levels of interaction. The low level component provides only essential access to the haptic device. The high level access component already offers sophisticated functionalities to facilitate the force calculation through an abstraction layer that hides the low level access. The conception of the haptic rendering pipeline requires the low level access to the underlying haptic device. The high level functionalities will not be used.

4.2 Prototype Development

The prototype is a set of object oriented C++ classes complementing the current source of the MarsVis. A smooth integration of the haptic functionality is not easily possible, because the MarsVis was not designed for comprehensive haptic support and is missing sophisticated interfaces. For an integration, the anchors for accessing the DTM, the scene graph (especially the platform node of the camera) and the input device interface has to be determined by a throughout investigation of the existing source code Those anchors needs to be modified to allow the exchange of information with the haptic expansion. The scheme in Figure 26 is a raw draft of the required additional components and the connection to the anchors. The real structure of classes may vary from this representation. New components are represented in blue, while the existing (modified) components are black.

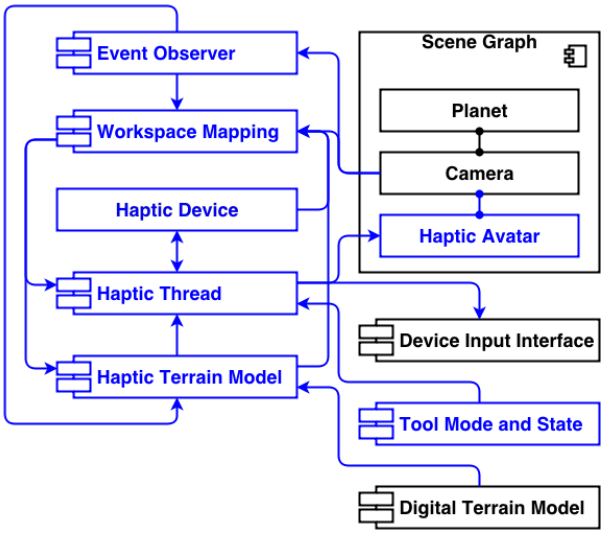


Figure 26: Outline of the implemented modules

The components of this draft should not be confused with a one-on-one representation of a corresponding class diagram. Rather these components should

be considered as logical connected modules to express the functionality and connectivity.

4.2.1 Haptic Device Interface

The haptic device module is the interface to the Phantom Omni haptic device. It allows to acquire and deliver input and output parameters such as position, rotation, various capability constants and forces. This module is build on the low level OpenHaptics API. It is plain and can be merged into the haptic thread module.

4.2.2 Haptic Avatar

The haptic avatar is a node in the existing scene graph, which is attached to the camera. It is hooked to the camera instead of the planet, because the camera coordinate system and the haptic device coordinate system are coincident. The graphical representation of the avatar node is a simple cone where the tip is the equivalent to the tip of the haptic stylus. A tailplane at the end of the cone visually indicates the current roll of the haptic stylus. Besides the position, that is set by the haptic thread, there is no other significant functionality.

4.2.3 Workspace Mapping

The workspace mapping module is a mandatory helper module, to perform various coordinate transformations back and forth between the different workspaces. This module stores the transformation states (matrices), which are exclusive for the current viewport. An update function allows to adjust the transformation states regarding a new viewport of the camera. The required parameters to update the transformation states, such as workspace dimensions, frustum angles, or existing transformations of nodes in the scene graph, are acquired from the haptic device module, the haptic terrain module and the camera module. Various workspace related mapping functions uses the stored transformation states to return the converted coordinates for the requested source coordinates. The workspace mapping module publicly exposes those functionalities and is accessible by all other modules.

4.2.4 Tool Mode and State

This module is a bridge to expose the current mode (navigation, profile liner, volume measurer) and state (list of lines/points) of the geodesy tool to the haptic thread. All geodesy tools are exclusively treated in the top level Python layer and not exposed or accessible within the MarsVis C++ library. The reason to introduce this module is to store the current tool mode and state of the Python layer and expose it to the haptic thread for virtual fixture processing.

4.2.5 Haptic Thread

The haptic rendering algorithm is executed in an encapsulated thread. It runs as separated loop along with the graphic rendering loop. The haptic thread is like the heart of the haptic expansion and pulls all the strings. The workspace module is used to perform the corresponding coordinate transformations. The position and button state input is acquired from the haptic device module. After updating the avatar with the current position, the button states in the device input interface are refreshed. The haptic terrain module offers the terrain data for the penetration force calculation. This module also processes the computation of forces for virtual fixtures based on the parameters from the tool mode and state module. The resulting force is delivered to the haptic device.

4.2.6 Haptic Terrain Model

The purpose of the module called haptic terrain model, is to store and provide access to the terrain data currently visible in the viewport and reachable within the workspace of the haptic device. The workspace mapping module is used to re-sample the stored terrain data from the DTM, when the event observer module notifies about a change of the viewport.

4.2.7 Event Observer

The event observer is responsible to watch for changes of the camera node in the scene graph. When a transformation of the camera node occurs, the workspace mapping will be notified. After the workspace mapping has been configured for the new viewport, the haptic terrain model needs also to be notified to update its content.

5 Evaluation

The main objective of the evaluation is to verify two hypotheses regarding the improvements induced by the added haptic support.

Theorem 5.1 *The tasks will be solved faster with the haptic device compared to a generic mouse.*

Theorem 5.2 *The results will have a higher accuracy with the haptic device compared to a generic mouse.*

Obviously the best way to achieve this is by comparing the workflow of the system in its original state with the workflow of the force feedback supported system. Both systems involve interfaces which are controlled by a user. Therefore an automated numerical analysis of runtime and accuracy can not be performed. Instead an empirical human subject pilot study will be conducted to evaluate the implemented concepts and designs.

5.1 Task Preparation

There are two implemented geodesy tools that will be tested with the mouse and the haptic device. For each geodesy tool a task should be performed with both devices. The users have to complete three repetitions for each device. To prevent learning effects, the device will be alternated after each repetition.

5.1.1 Profile Liner Assignment

The first task is focused on the workflow with the profile liner. A clearly visible canyon from a perspective view is presented to the user. The user has to draw five lines with particular requirements across the canyon. All lines have to be perpendicular to the orientation of the canyon. The start and endpoints of the lines must be aligned to the start and endpoint of the first line which acts as the reference line. Furthermore, all lines should be parallel. The task should be performed precisely and quickly. An exemplary illustration was given in Figure 7 in the introduction.

5.1.2 Volume Measurer Assignment

The second task is meant for the volume measurer. A clearly visible crater from a perspective view is presented to the user. In order to calculate the inner volume of the crater, a bounding poly-line must be drawn along the surrounding rim. The points of the poly-line should be placed as close to the mountain ridge as possible. An exemplary illustration was given in Figure 8 in the introduction. The number of the points or the distance between the points should roughly be the same for all repetitions. The task should be performed precisely and quickly.

5.2 Data Acquisition

A meaningful evaluation of the results requires a thorough consideration of the information that has to be acquired.

5.2.1 Measured Data

An important value that is logged in all tests is the update rate of the haptic rendering pipeline. The time will be captured for each repetition.

For the profile liner the time excludes the duration for drawing the first line, since the time is dedicated for orthogonal adjustment across the canyon. Furthermore the coordinates of the start and endpoint for each line will be recorded. The lines defined by these points will be investigated for their deviation in length and orientation. The orientation is expressed as the angle between a line and an independent reference line. Related to the location of the given canyon, the x-axis is sensitive to angular differences between the lines and will be used as reference line.

For the volume measurer the timespan starts by setting the first point and ends by closing the poly-line. The capacity will be stored to analyze the standard deviations for each device. A reliable reference capacity of the chosen crate does not exist. The used reference capacity of $-4.50441 \times 10^{12}\ m^3$ was determined in advance using the highest resolution of the terrain data and other aids such as force feedback, height amplification and wireframe visualization. A negative capacity indicates that the amount of material is missing.

5.2.2 Subjective User Data

Another objective of this pilot study is to acquire subjective impressions of the individual users. To assess the individual feedback, it is mandatory to expose the current experience level of each user. Especially information about the ability to utilize the Virtual Reality terrain visualization and exploration framework (MarsVis) and kinesthetic haptic devices is desired. A point of interest after the completion of both assignments is the preferred device for each task, e.g. which device delivers a superior feeling of confidence and efficiency. Furthermore, the general attitude of the users toward kinesthetic haptic devices will be determined for each user. The interview is also focused on the opinions related to the tasks and the devices, especially the support or obstruction of the virtual fixtures.

5.3 Environment

All tests will be performed in a separate office. The desktop is built around a Dell workstation with an Intel Xeon 5500 i7 @ 2.4GHz and 24GB RAM, see Figure 27. A generic mouse and a Phantom Omni haptic device are used as the

input devices. The visual output is rendered through a nVidia Quadro FX5800 graphic adapter on a 23.6" screen. The rendered force output will be redirected to the Phantom Omni haptic device. SuSE Linux Enterprise Desktop 11.2 is deployed as operating system. As stated previously, the MarsVis is the software application that will be used in combination with the dataset from the Mars.

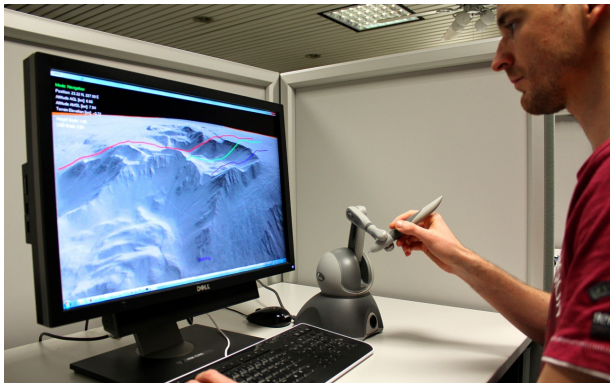


Figure 27: Evaluation workplace in the laboratory

The user input depends on the specific software mode. The mode without haptic support facilitates only the mouse to interact with the geodesy tools. The mode with haptic support exclusively uses the haptic device to control the geodesy tools.

5.4 Participants

Regarding the context of this work, the participants should be from the planetary research domain with experience in terrain exploration and investigation. Unfortunately it was impossible to schedule an appointment within the timeframe. In place of the planetary researchers, male and female students and researchers from the field of computer science participated in the study. The age ranges roughly between 25 and 40 years. Only a few participants have experience with Geographic Information Systems (GISs) or the MarsVis and commercial kinesthetic haptic devices. All users are in a healthy condition without any visual, tactile or kinesthetic disabilities. A total amount of eight users participated in the study.

5.5 Procedure

After a brief introduction into the project and the assignments, each user will be informed about the safety notes for this evaluation with the opportunity to refuse the participation. The participating users get a detailed explanation of the tasks that should be performed. A guided tutorial with trial attempts will ensure that the user is capable to control the geodesy tools with the mouse

and the haptic device. Subsequently the user starts to solve the given assignments. After all tasks have been completed, the impressions of the user are discussed and recorded in a concluding interview under guidance of a prepared questionnaire (Appendix A.1).

5.6 Results

Regarding their previous experience, two groups emerge from the participants. The first group consists of two participants. Both of them have already used the MarsVis and kinesthetic haptic devices. The users of the second group are completely inexperienced.

5.6.1 Measured Results

The pilot study was conducted with a small number of participants which performed only a few repetitions. The amount of measured samples from the profile liner and volume measurer tasks is insufficient to draw representative statistic conclusions. The presented average values and standard deviations are only addressed to get a first impression and figure out trends. All stochastic calculations are based on the normal distribution, since a related distribution function can not be determined with the small amount of samples.

Haptic Rendering Performance

The update rate of the haptic rendering pipeline satisfies the expectations and was fast enough to keep up with the capabilities of the Phantom Omni haptic device, see Figure 28. The average frequency during the tests was 996 Hz with a standard deviation of 7 Hz. In the volume measurer mode the frequency dropped to its lowest recorded value of 940 Hz, the maximum value of 998 Hz was reached about 93% of the time and is obviously limited by the haptic device.

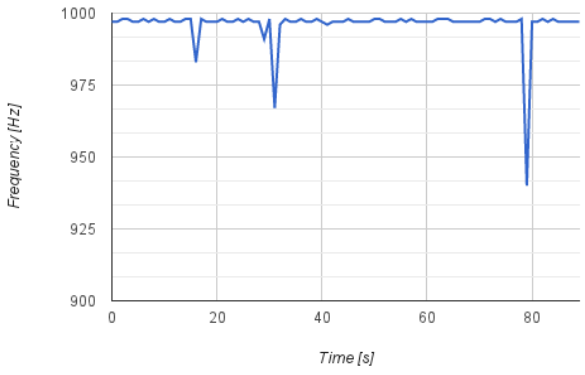
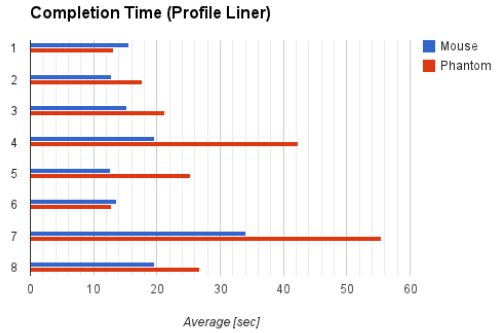


Figure 28: Update rate of the haptic rendering pipeline (extract)

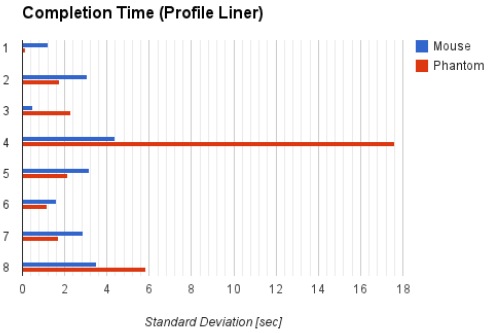
Pofile Liner Assignement

The inexperienced participants were able to solve the task much faster with the mouse, while the experienced users performed well with both devices. see Figure 29. Each repetition offers a standard deviation for the lengths of the lines and the angles of the lines. These standard deviations have been averaged over the repetitions for each input device. It is noticeable that the lines drawn with the haptic device have a lower deviation in their length and angles compared to the lines drawn with the mouse. Exceptions are the third user, who refuses the adaption of the haptic device and the seventh user, who trades completion time for accuracy (Appendix A.2).

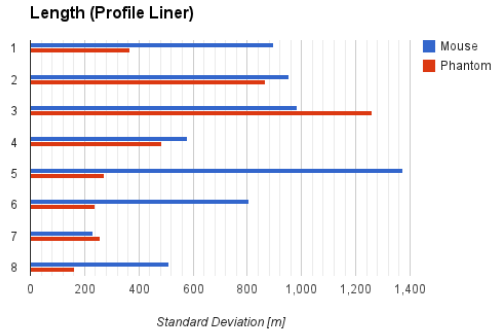
Hypothesis 5.1 was not met, the completion time seems to be directly affected by the ability of the user to utilize the input device. It is assumed that the completion time will be reduced when the users are intensively trained with the operation of the haptic device. The second Hypothesis 5.2 that predicted a higher accuracy with the haptic device was confirmed by the majority of the participants. The results from the tests with the haptic device have a higher accuracy regarding the requirements of the task, than the results from the tests with the mouse. This comparison refers to the current state of the MarsVis, which has no visual fixtures or pseudo haptic effects. It should be considered that such effects will probably improve the accuracy for the mouse driven interaction.



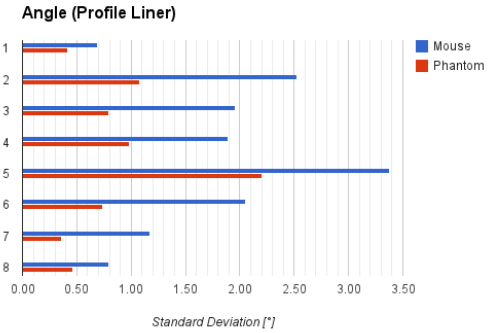
(a) Average completion time



(b) Standard deviation of completion time



(c) Standard deviation of arc length



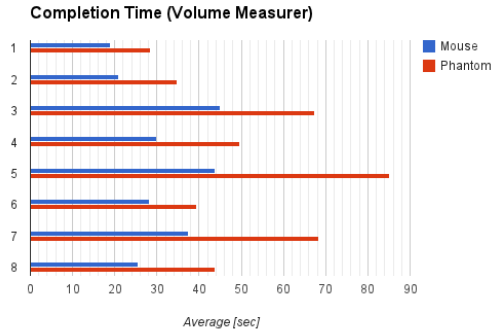
(d) Standard deviation of angle

Figure 29: Overview of the measurements from the profile liner assignment for each participant

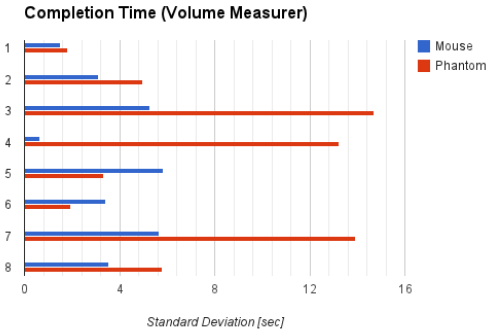
Volume Measurer

All users were able to perform the given task faster with the mouse interface than with the haptic device, see Figure 30. The difference of the standard deviations of the measured capacity between the input devices varied. Some users reached a lower capacity deviation when using the haptic device, while others achieved better results with the mouse. It is also impossible to draw conclusions considering the difference between the measured capacity and the reference capacity (Appendix A.2).

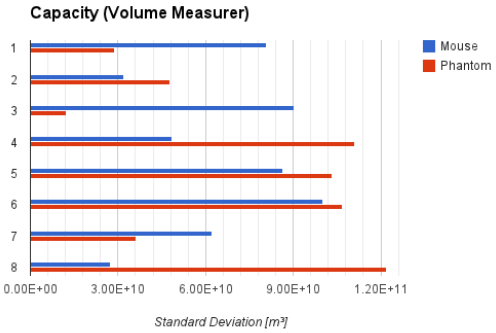
Analogous to the profile liner, Hypothesis 5.1 was not met. It is still assumed that the completion time of the inexperienced users will be reduced by additional usability training with the haptic device. Furthermore it was assumed in Hyothesis 5.2 that the haptic guidance will simplify the determination of the accurate poly-line surrounding the crate. It was shown by the results, that the concept of the virtual fixture for the volume measurer did not bring the expected benefit.



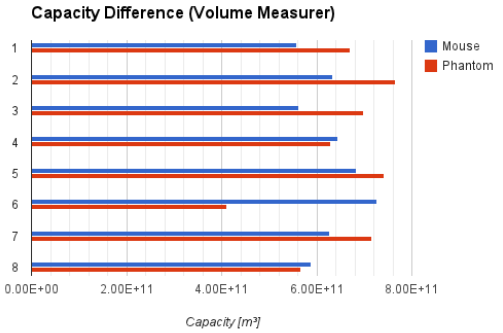
(a) Average completion time



(b) Standard deviation of completion time



(c) Standard deviation of capacity



(d) Difference between average and reference capacity

Figure 30: Overview of the measurements from the volume measurer assignment for each participant

5.6.2 User Feedback

All users reported the ease of use when solving the tasks with the mouse, but also mentioned their doubts about the accuracy. Apart from a single user who completely refuses the adaption of the haptic device, the general attitude regarding the usage of haptic devices was open minded. The participants would consider the usage of such devices for other respective applications. On the other hand the handling of the haptic device was quite difficult and inconvenient for the inexperienced users, in particular:

- Operating the floating end effector felt inaccurate without a solid ground such as the desktop surface for the mouse
- The ability to move the device in three spatial dimensions was unfamiliar
- The two dimensional projection of the avatar on the screen was not giving the required depth information (hand-eye coordination problems)

As a result of this, some users unconsciously relied primarily on the visual information and often lost contact with the terrain surface. In general the users found the additional snapping helpful for both tools. Some users disliked to sense the discontinuities of the terrain surface while drawing the profile lines and would prefer to feel the flat surface of a sphere or a plane. In case of the volume measurer the guiding contour around the crate that was inconsistent. Some points along the contour had a higher snapping and trapped the user, which limited the degree of freedom when placing the points and prevents the user from a smooth movement. Another issue that was monitored during the volume measurer assignment was a random hang-up in a vibration loop. This behavior often occurred when the user pushed tightly onto the terrain near snapping points. It was also noted that the visual appearance of the terrain at the avatar position does not match the perceived forces. Previous investigations of the chosen regions from different perspectives and levels of detail were made to confirm that the calculated forces coincides with the Digital Terrain Model (DTM). The users were confused again by prioritizing the sense of sight over the sense of touch. A minority of the users were dissatisfied with the small size of the workspace and the low magnitude of the force response.

6 Conclusion

This work presented the integration of force feedback into the planetary terrain exploration and analysis. In this context the sense of touch was added to provide the feeling of discontinuities of the planet's surface. Furthermore two geodesy analysis tools were complemented by virtual fixtures to improve the workflow. The final evaluation has shown that a clear assessment of the presented assumptions and concepts is not possible, but the pilot study revealed some tendencies, strengths and weaknesses.

6.1 Summary

The developed haptic rendering pipeline operated fast enough to keep up with the maximum refresh rate of the used Phantom Omni haptic device. The prototype implementation does not facilitate any computational optimizations such as parallelization or shader processing on the GPU, which leaves some room for further improvements. Rendering forces within a vast amount of terrain data was successfully demonstrated.

The stiffness of the terrain surface can be further increased with a haptic device that provides a higher stiffness limit such as the Force Dimension Omega 6. The hardness might improve the realistic feeling of the surface and the terrain features. However, there is no proof that results from the evaluation can be improved when the terrain felts harder.

It was expected that the realistic terrain surface will assist in the precise placement of the profile lines. In opposition to this assumption the pilot study exposed that the workflow of the profile liner could be enhanced by simulating a flat terrain surface during the drawing operation. The force-based snapping improved the accuracy compared to the results achieved with the mouse. On the other hand the performance with the mouse was faster. It is still impossible to determine which device is more efficient, the workflow of each device can be improved with additional usability training or features such as pseudo/visual haptics.

Based on the varied measurements and the given user feedback, it is precarious to make conclusions related to the volume measurer. The concept and implementation of the virtual fixture that should assist the user to detect the outline of the crate did not produce the desired results. The terrain feature detection and the creation of the flux values needs to be improved. Flooding or other algorithms should deliver a more reliable terrain feature detection with less noise. The differences of connected flux must be small to prevent from being trapped when moving along.

6.2 Future Work

This work was focused on haptic rendering and paid no further attention to the hand-eye coordination or to the visual representation of the avatar. With respect to the pilot study, these aspects have the capability for further improvement of the user interaction. In a sequential work the visualization techniques for haptic interfaces could be investigated. Techniques such as display mirroring, stereoscopic rendering or artificial light and shadows may greatly improve the hand-eye coordination and depth impression.

The presented terrain feature detection was based on an edge detection from the image processing domain. This was a fast and simple technique to form a base for the development of the virtual fixture for the volume measurer, but it is not reliable for continuous terrain changes. A follow-up work could discuss various terrain detection algorithms such as flooding. Furthermore an appropriate procedure could be developed to create a consistent and smooth flux map from the terrain features. Or even better investigate procedures to extract splines or trails from the terrain features, which can be traced by haptic guidance.

The pilot study exposed plenty of interesting thoughts that could be considered further. The concepts of this work could be revised and improved regarding the results from the evaluation. A comprehensive user study could be prepared with assignments and questions pinpointing and isolating the various impressions from the pilot study. Also the participants for this study should be trained in advance to get used to the haptic device. The results from this study should be more significant to distinguish the benefit and improvement gained by force feedback assistance.

Two geodesy tools have been enhanced with force feedback so far. Another interesting and commonly used tool is the navigator. The navigation in the desktop version of the Virtual Reality terrain visualization and exploration framework (MarsVis) is still performed with the mouse. A future work could investigate the possibilities of a navigation based on a six degrees of freedom haptic device. This work could also consider aspects of a force feedback supported navigation.

References

[1] O'Malley, M.K. & Gupta, A. *"HCI Beyond the GUI: Design for Haptic, Speech, Olfactory, and Other Nontraditional Interfaces"*, Morgan Kaufmann Publishers Inc. San Francisco, CA, USA, pages 25-74, 2008

[2] Akamastu, M., MacKenzie, I.S. & Hasbrouq, T. *"A Comparsion of Tactile, Auditory, and Visual Feedback in Pointing Task Using a Mouse-Type Device"*, Ergonomics, 38, 816-827, 1995

[3] Ng, A.W.Y. & Chan, A.H.S. *"Finger Response Times to Visual, Auditory and Tactile Modality Stimuli"*, Proceedings of the IMECS, Vol. II, 2012

[4] Peon A.R. & Prattichizzo, D. *"Reaction times to constraint violation in haptics: comparing vibration, visual and audio stimuli"*, IEEE World Haptics Conference (WHC), pages 657-651, 2013

[5] Rosenberg, L.B. *"Virtual fixtures: Perceptual tools for telerobotic manipulation"*, IEEE Virtual Reality Annual International Symposium, pages 76-82, 1993

[6] Gunn C., Muller, W. & Datta, A. *"Performance Improvement with Haptic Assistance: A Quantitative Assessment"*, Third Joint Eurohaptics Conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems, Vol. 1, pages 511 - 516, 2009

[7] Coles, T.R., Meglan, D. & John, N.W. *"The Role of Haptics in Medical Training Simulators: A Survey of the State of the Art"*, IEEE Transactions on Haptics, Vol. 4, No. 1, pages 51-66, 2011

[8] Hannaford, B. & Wood, L. *"Performance evaluation of a 6 axis high fidelity generalized force reflecting teleoperator"*, The JPL/NASA Conf. Space Telerobotics, pages 89-97, 1989

[9] Preusche, C. & Hirzinger, G. *"Haptics in telerobotics: Current and future research and applications"*, The Visual Computer: International Journal of Computer Graphics, Vol. 23, Issue 4, pages 273-284, 2007

[10] Faeth, A., Oren, M. & Harding, C. *"Combining 3-D geovisualization with force feedback driven user interaction"*, ACM GIS, Article No. 25, 2008

[11] Raghupathy, P.B. & Borst, C.W. *"Force Feedback and Visual Constraint for Drawing on a Terrain: Path Type, View Complexity, and Pseudo-haptic Effect"*, 3DUI 2012, pages 157-158, 2012

[12] Buckley, D.J. *"The GIS Primer – An Introduction to Geographic Information Systems"*, 1997

[13] *"The GIS Encyclopedia"*, Wiki.GIS.com, `http://wiki.gis.com`, Nov. 21, 2013

[14] Folgert, J. *"A Short Introduction to GIS"*, KnowGIS, `http://www.knowgis.com/free_knowgis.html`, Nov. 17, 2013

[15] *"GIS Dictionary: GIS"* ESRI, `http://support.esri.com/en/knowledgebase/GISDictionary/term/GIS`, Nov. 23, 2013

[16] Salisbury, J.K., Barbagli, F. & Conti, F. *"Haptic Rendering: Introductory Concepts"*, IEEE Computer Graphics and Applications, Vol. 24, No. 2, pages 24-32, 2004

[17] Salisbury, J.K., Brock, D., Massie, T., Swarup, N. & Zilles, C. *"Haptic rendering: Programming touch interaction with virtual objects"*, Proceedings of the Symposium on Interactive 3D Graphics, pages 123-130, 1995

[18] Zilles, C.B. & Salisbury, J.K. *"A Constraint-based God-object Method For Haptic Display"*, Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, pages 146-151, 1995

[19] McNeely, W.A., Puterbaugh, K.D. & Troy, J.J. *"Six Degree-of-Freedom Haptic Rendering Using Voxel Sampling"*, Proceedings of ACM SIGGRAPH, pages 401-408, 1999

[20] Gregory, A., Lin, M.C., Gottschalk, S., & Taylor, R. *"A Framework for Fast and Accurate Collision Detection for Haptic Interaction"*, IEEE Virtual Reality, pages 38-45, 1999

[21] Kim, L., Kyrikou, A., Sukhatme, G.S. & Desbrun, M. *"An Implicit-Based Haptic Rendering Technique"*, IEEE/RSJ International Conference Intelligent Robots and Systems, Vol. 3, pages 2943-2948, 2002

[22] Ho, C.-H., Basdogan, C. & Srinivasan, M.A. *"Efficient Point-Based Rendering Techniques for Haptic Display of Virtual Objects"*, Presence: Teleoperators and Virtual Environments, Vol. 8, No. 5, pages 477–491, 1999

[23] Conti, F. & Khatib, O. *"Spanning large workspaces using small haptic devices"*, Proceedings of the First Joint Eurohaptics Conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems, pages 183-188, 2005

[24] Tatematsu, A. & Ishibashi, Y. *"Mapping Workspaces to Virtual Space in Work Using Heterogeneous Haptic Interface Devices"*, Advances in Haptics, InTech, pages 621-636, 2010

[25] Bau, O., Poupyrev, I., Israr, A. & Chris Harrison, C. *"TeslaTouch: Electrovibration for Touch Surfaces"*, Proceedings of the 23nd Annual ACM Symposium on User Interface Software and Technology, pages 283-292, 2010

[26] Kajimoto, H., Kawakami, N. Tachi, S. & Inami, M. *"SmartTouch: Electric Skin to Touch the Untouchable"*, IEEE Computer Graphics and Applications, Vol. 24, No. 1, pages 36-43, 2004

[27] Hoshi, T., Takahashi, M., Iwamoto, T., & Shinoda, H. *"Noncontact Tactile Display Based on Radiation Pressure of Airborne Ultrasound"*, IEEE Transactions on Haptics, Vol. 3, 155 - 165, 2010

[28] Gupta, S., Morris, D., Patel, S.N. & Tan, D. *"AirWave: Non-contact Haptic Feedback Using Air Vortex Rings"*, Proceedings of the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing, pages 419-428 2013

[29] Berkelman, P. & Dzadovsky, M. *"Using Magnetic Levitation for Haptic Interaction, Advances in Haptics"*, Advances in Haptics, InTech, pages 31-46, 2010

[30] Williams II, R.L. *"Cable-Suspended Haptic Interface"*, International Journal of Virtual Reality, Vol. 3, No. 3, pages 13-21 1998

[31] Koul, M. H., Kumar, P., Singh, P. K., Manivannan M. & Saha, S. K. *"Gravity Compensation for Phantom Omni Haptic Interface"*, First Joint International Conference on Multibody System Dynamics 2010

[32] Górski, K.M., Wandelt, B.D., Hivon, E., Hansen, F.K. & Banday, A.J. *"The HEALPix Primer"*, eprint, arXiv, astro-ph/9905275, 1999

[33] Pajarola, R. *"Overview of Quadtree-based Terrain Triangulation and Visualization"*, UCI-ICS Technical Report, No. 02-01, 2002

[34] Westerteiger, R., Gerndt, A., Hamann, B. & Hagen, H. *"Spatial Analysis of Terrain in Virtual Reality"*, IEEE Virtual Reality Workshop, Immersive Visualization Revisited: Challenges and Opportunities, 2012

[35] Westerteiger, R. *"Cartography of Mars in a Virtual Reality Environment"*, VLUDS, Vol. 19, pages 100-110, 2010

[36] Kuhlen, T., Beer, T., & Gerndt, A. *" The ViSTA Virtual Reality Toolkit"*, Proceedings of the 5th High-End Visualization Workshop, Baton Rouge, Louisiana, page 4, 2009

A Appendix

A.1 Questionnaire Template

MarsVis Guidance/Questionnaire (Interview based, filled by instructor, no handouts for the participants)

The objective of this user study is to test certain tasks performed in geodesy analysis with two different interface devices. The software environment for these tests will be MarsVis, a VR framework to explorer and investigate the surface of the Mars.

Safety Notes

The Phantom Omni is a haptic device that relays forces to the operator. Forces may cause physical injuries. The Phantom Omni is only capable to render forces up to three Newton which can theoretically be assumed as harmless.

Intensive colors and virtual flights combined with a stereoscopic display may lead to visual overstimulation.

General

1. **Gender**
   Mark only one oval.
   - ◯ Male
   - ◯ Female

2. **Age**

   .....

3. **Occupation**

   .....

4. **User Experience**
   Rate your experience with the specific hardware/software components
   Mark only one oval per row.

	None	Basic	Intermediate	Advanced
Geographic Information Systems	◯	◯	◯	◯
MarsVis	◯	◯	◯	◯
Desktop Mouse	◯	◯	◯	◯
Haptic Stylus	◯	◯	◯	◯

Task: Profile Liner

The user has to draw five profile lines across the given canyon. The lines should be parallel, with aligned start-points and endpoints (preferable equal length). This task has to be completed six times in a row, alternating between mouse and haptic device.

GUIDED TUTORIAL:

* drawing profile lines with mouse & phantom
* setting start/endpoints
* remove line(s)
* confirm a set of profiles
* questions?

1. **Mouse (Repetition 1)**

   .....

   Example: 4:03:32 (4 hours, 3 minutes, 32 seconds)

2. **Haptic (Repetition 2)**

   .....

   Example: 4:03:32 (4 hours, 3 minutes, 32 seconds)

3. **Mouse (Repetition 3)**

   .....

   Example: 4:03:32 (4 hours, 3 minutes, 32 seconds)

4. **Haptic (Repetition 4)**

   .....

   Example: 4:03:32 (4 hours, 3 minutes, 32 seconds)

5. **Mouse (Repetition 5)**

   .....

   Example: 4:03:32 (4 hours, 3 minutes, 32 seconds)

6. **Haptic (Repetition 6)**

   .....

   Example: 4:03:32 (4 hours, 3 minutes, 32 seconds)

7. **Results from logged data**

   .....

   .....

   .....

   .....

   .....

Task: Volume Measurer

The user has to draw a poly-line around the given crate. The points for the poly-line should be placed topmost on the surrounding rim. This task has to be completed six times in a row, alternating between mouse and haptic device.

GUIDED TUTORIAL:

* drawing volume poly-line with mouse & phantom
* adding points to the poly-line
* confirm a volume
* questions?

1. **Mouse (Repetition 1)**

   .....

   Example: 4:03:32 (4 hours, 3 minutes, 32 seconds)

2. **Haptic (Repetition 2)**

   .....

   Example: 4:03:32 (4 hours, 3 minutes, 32 seconds)

3. **Mouse (Repetition 3)**

   .....

   Example: 4:03:32 (4 hours, 3 minutes, 32 seconds)

4. **Haptic (Repetition 4)**

   .....

   Example: 4:03:32 (4 hours, 3 minutes, 32 seconds)

5. **Mouse (Repetition 5)**

   .....

   Example: 4:03:32 (4 hours, 3 minutes, 32 seconds)

6. **Haptic (Repetition 6)**

   .....

   Example: 4:03:32 (4 hours, 3 minutes, 32 seconds)

7. **Results from logged data**

   .....

   .....

   .....

   .....

   .....

User Feedback

Interview with the participant (focus on the following points of discussion)

1. **Which device would you exclusive choose if you have to solve the tasks again?**
   Mark only one oval.

   - ◯ Mouse for profile liner and volume measurer
   - ◯ Haptic device for profile liner and volume measurer
   - ◯ Mouse for profile liner and haptic device for volume measurer
   - ◯ Haptic device for profile liner and mouse for volume measurer

2. **How do you assess the ease of completing the tasks?**
   Overall rating

3. **Was the usability of the haptic device pleasant?**
   Rating related to the haptic hardware

4. **Was the haptic guidance helpful to complete the tasks?**
   In which particular situation?

5. **Was the haptic guidance obstructive in the workflow?**
   In which particular situation?

6. **List of most negative and positive aspects**
   User specific list of likes and dislikes

A.2 Measurements (Overview)

	User 1	User 2	User 3	User 4	User 5	User 6	*User 7	User 8
EXPERIENCE								
GIS	1/3	2/3	0/3	0/3	0/3	1/3	1/3	1/3
MarsVis	3/3	0/3	0/3	0/3	0/3	3/3	0/3	1/3
Mouse	3/3	3/3	3/3	3/3	3/3	3/3	3/3	3/3
Haptic	2/3	0/3	0/3	0/3	0/3	2/3	0/3	1/3
PROFILE LINER								
Mouse Time 1	16.40	16.40	n.a.	20.90	10.90	15.40	37.40	22.80
Mouse Time 2	16.00	10.90	14.90	14.70	10.70	13.10	32.20	20.30
Mouse Time 3	14.10	11.30	15.60	23.20	16.30	12.30	32.60	15.90
Mouse Time AVG	**15.50**	**12.87**	**15.25**	**19.60**	**12.63**	**13.60**	**34.07**	**19.67**
Mouse ArcLength (Averaged StdDev)	896.00	954.00	983.00	580.00	1374.00	805.00	232.00	512.00
Mouse Angle [°] (Averaged StdDev)	0.69	2.53	1.96	1.89	3.38	2.05	1.17	0.79
Haptic Time 1	13.00	15.80	20.10	37.20	26.10	14.00	53.70	25.50
Haptic Time 2	13.20	19.30	23.90	47.60	27.00	12.90	57.10	33.20
Haptic Time 3	13.20	18.00	19.80	71.50	22.90	11.70	55.80	21.70
Haptic Time AVG	**13.13**	**17.70**	**21.27**	**42.40**	**25.33**	**12.87**	**55.53**	**26.80**
Haptic ArcLength (Averaged StdDev)	367.00	865.00	1259.00	485.00	274.00	238	256.00	164.00
Haptic Angle [°] (Averaged StdDev)	0.42	1.08	0.79	0.98	2.20	0.74	0.36	0.46
VOLUME MEASURER								
Mouse Time 1	17.20	24.50	51.00	30.60	37.00	24.70	34.30	29.70
Mouse Time 2	19.60	18.50	43.10	30.20	47.20	31.50	43.90	23.60
Mouse Time 3	20.00	20.00	41.00	29.30	47.00	28.60	33.90	23.50
Mouse Time AVG	**18.93**	**21.00**	**45.03**	**30.03**	**43.73**	**28.27**	**37.37**	**25.60**
Mouse Volume 1	-4.04E+12	-3.90E+12	-3.84E+12	-3.81E+12	-3.81E+12	-3.83E+12	-3.82E+12	-3.94E+12
Mouse Volume 2	-3.88E+12	-3.84E+12	-3.98E+12	-3.91E+12	-3.91E+12	-3.66E+12	-3.94E+12	-3.89E+12
Mouse Volume 3	-3.93E+12	-3.88E+12	-4.00E+12	-3.86E+12	-3.74E+12	-3.84E+12	-3.88E+12	-3.92E+12
Haptic Time 1	26.60	35.30	84.10	41.80	88.90	37.30	60.30	49.80
Haptic Time 2	28.30	39.30	56.40	42.00	82.40	41.10	60.10	43.60
Haptic Time 3	30.20	29.40	61.70	64.80	84.40	39.80	84.30	38.20
Haptic Time AVG	**28.37**	**34.67**	**67.40**	**49.53**	**85.23**	**39.40**	**68.23**	**43.87**
Haptic Volume 1	-3.80E+12	-3.79E+12	-3.80E+12	-3.82E+12	-3.65E+12	-3.97E+12	-3.77E+12	-3.90E+12
Haptic Volume 2	-3.84E+12	-3.70E+12	-3.82E+12	-3.81E+12	-3.79E+12	-4.16E+12	-3.77E+12	-4.07E+12
Haptic Volume 3	-3.86E+12	-3.72E+12	-3.80E+12	-4.00E+12	-3.84E+12	-4.15E+12	-3.83E+12	-3.84E+12
USER PREFERENCE								
Profile Liner	Haptic	Mouse	Mouse	Haptic	Haptic	Haptic	Haptic	Mouse
Volume Measurer	Mouse	Haptic	Mouse	Mouse	Mouse	Haptic	Haptic	Haptic

*used the visible grid for assistance
Reference Volume (based on thoroughly measurement): -4.50441E+12