

TECHNISCHE UNIVERSITÄT BERLIN

MASTER THESIS

---

**Analysis of illumination conditions at  
the lunar south pole using parallel  
computing techniques**

---

*Author:*

Ramiro MARCO FIGUERA

*Supervisors:*

Dipl.-Ing. Philipp GLÄSER

Prof. Dr. Jürgen OBERST

*A thesis submitted in fulfilment of the requirements  
for the degree of Master of Science*

*in the*

Planetary Geodesy Research Group  
Department of Geodesy and Geoinformation Science

November 2014

# Declaration of Authorship

I, Ramiro MARCO FIGUERA, declare that this thesis titled, 'Analysis of illumination conditions at the lunar south pole using parallel computing techniques' and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed:

---

Date:

---

TECHNISCHE UNIVERSITÄT BERLIN

## *Abstract*

Faculty VI Planning - Building - Environment  
Department of Geodesy and Geoinformation Science

Master of Science

### **Analysis of illumination conditions at the lunar south pole using parallel computing techniques**

by Ramiro MARCO FIGUERA

In this Master Thesis an analysis of illumination conditions at the lunar south pole using parallel computing techniques is presented. Due to the small inclination ( $1.54^\circ$ ) of the lunar rotational axis with respect to the ecliptic plane and the topography of the lunar south pole, which allows long illumination periods, the study of illumination conditions is of great importance. As a result of the large amount of computation time needed to calculate illumination conditions with sequential programming techniques, a new parallel tool was developed. Different tests are presented using different areas and periods of time. The cross-validation of the tool with TUB-Shader and ICAT is presented. Two case studies analyzing the landing site candidate for the Russian Luna-Resurs and the Shackleton Vicinity (SV1) landing site are presented. This tool allowed us to conduct simulations in a reduced amount of time. As an example we have reduced the computation time from a 24 hours process analyzing a 20 km by 20 km region over a 1 year period to 25 minutes for the same set up.

## *Acknowledgements*

I would like to express my dearest appreciation to Prof. Jürgen Oberst, Philipp Gläser and Diego de Rosa for the opportunity they offered me to work with them and for their time and supervision. Philipp Gläser has provided me the data for this research and, more important, advice and counseling when I was lost, for that, I am very thankful to him. I want to thank Diego de Rosa for giving me the opportunity to work with his team at ESA for 3 months. I would never forget my time there. I would also thank all the people that has crossed with me during this research. My very special thanks to all my friends here in Berlin, Barcelona and the rest of the world. Thanks to Xavier Lopez Llauredó, Mar Ortega de la Plaza, Bernat Romagosa Carrasquer and Marc Martinez Gargallo for sending me energy from the distance. Thanks to my family for their unconditional support and, last but not least, my most special thanks to my lovely girlfriend Marta. Gràcies per ajudar-me a anar traient les pedres del camí. Thanks, danke, gràcies, gracias.



# Contents

<b>Declaration of Authorship</b>	<b>i</b>
<b>Abstract</b>	<b>ii</b>
<b>Acknowledgements</b>	<b>iii</b>
<b>Contents</b>	<b>iv</b>
<b>List of Figures</b>	<b>vi</b>
<b>List of Tables</b>	<b>viii</b>
<b>Abbreviations</b>	<b>ix</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Motivation . . . . .	1
1.2 Objectives . . . . .	2
1.3 Related work . . . . .	3
<b>2 The Moon</b>	<b>6</b>
2.1 Moon overview . . . . .	6
2.2 Lunar south pole . . . . .	7
2.3 Lunar lander missions . . . . .	8
<b>3 Methodology</b>	<b>11</b>
3.1 Data . . . . .	11
3.2 Computation methods . . . . .	12
3.2.1 Ray-tracing method . . . . .	15
3.2.2 Horizon method . . . . .	15
3.2.3 Illumination computation . . . . .	17
<b>4 Programming techniques</b>	<b>24</b>
4.1 Sequential programming . . . . .	24
4.2 Parallel programming . . . . .	25
4.2.1 Task parallelism . . . . .	26
4.2.2 Data Parallelism . . . . .	27
4.2.3 Introduction to OpenCL . . . . .	28

---

4.2.4	CPU vs. GPU . . . . .	29
<b>5</b>	<b>TUB-Light a Parallel programming illumination tool</b>	<b>31</b>
5.1	Overview . . . . .	31
5.2	Motivation . . . . .	32
5.3	Data . . . . .	32
5.4	Results . . . . .	33
5.4.1	2 x 2 km results . . . . .	34
5.4.2	20 x 20 km results . . . . .	36
5.4.3	50 x 50 km results . . . . .	39
5.4.4	Cross-validation with TUB-Shader . . . . .	41
<b>6</b>	<b>Case Study I: Illumination studies for Luna-Resurs mission</b>	<b>48</b>
6.1	Description of the Russian lunar mission: Luna-Resurs . . . . .	48
6.2	Data . . . . .	51
6.3	Results . . . . .	51
<b>7</b>	<b>Case Study II: Illumination conditions at Shackleton Vicinity candi- date Landing site</b>	<b>54</b>
7.1	Description of Shackleton Vicinity candidate Landing site . . . . .	54
7.2	Data . . . . .	55
7.3	Results . . . . .	55
<b>8</b>	<b>Conclusions and Outlook</b>	<b>60</b>
<b>A</b>	<b>TUB-Light: User manual</b>	<b>62</b>
A.1	Requirements . . . . .	62
A.2	Configuration Files . . . . .	62
A.2.1	Structure of Configuration Files . . . . .	63
A.3	Execution . . . . .	64
<b>B</b>	<b>Developers Manual</b>	<b>65</b>
B.1	APIs, libraries and toolkits used . . . . .	65
B.1.1	OpenCL . . . . .	65
B.1.2	Spice Toolkit . . . . .	65
B.1.3	GMT . . . . .	65
B.1.4	TUB-Light in double precision configuration . . . . .	66
	<b>Bibliography</b>	<b>67</b>

# List of Figures

2.1	The rotation, precession and nutation movement of the Moon. . . . .	6
2.2	Inclination of the Moon with respect to the ecliptic plane. . . . .	7
3.1	Comparison of NAC, LOLA and adjusted LOLA DTM. . . . .	12
3.2	Basic example of the calculation of the elevation for a terrain feature and for the Sun. . . . .	13
3.3	Vectorial solution scenario with the 3 initial vectors. . . . .	14
3.4	Vectors involved in the calculation of the elevations. . . . .	14
3.5	Calculation of the amount of visible Sun from a certain location . . . . .	18
3.6	Calculation of the elevations using vector algebra. . . . .	20
3.7	Value assignation for the non-intersecting cases. . . . .	21
3.8	Calculation of the visible circular segment. . . . .	22
3.9	Calculation of the wrong visible area . . . . .	23
4.1	Sequential Programming process . . . . .	24
4.2	Example of an IBM supercomputer . . . . .	25
4.3	Graphical example of task parallelism. . . . .	27
4.4	Graphical example of data parallelism. . . . .	28
4.5	Core comparison of the CPU and the GPU . . . . .	29
4.6	CPU vs GPU vector addition computation time comparison. . . . .	29
5.1	Gnomonic projection. . . . .	32
5.2	Computation time for all the tests. . . . .	33
5.3	Accumulated Illumination for an area of 2 x 2 km centered at CR1 over a one-year period at 2 m height. . . . .	34
5.4	Illumination pattern over a one-year period at 2 m height. . . . .	34
5.5	Illumination pattern located over a 20 year period. . . . .	35
5.6	Trajectory of the Sun and shape of the horizon. . . . .	36
5.7	Accumulated illumination for an area of 20 x 20 km at the lunar south pole over a one-year period at 2 m height. . . . .	37
5.8	Accumulated Illumination for an area of 50 x 50 km at the lunar south pole over a one-year period at 2 m height. . . . .	39
5.9	Trajectory of the Sun and shape of the horizon. . . . .	40
5.10	Illumination pattern for a location over a one-year period at 2 m height. . . . .	41
5.11	Comparison of illumination differences using floating points and double precision. . . . .	44
5.12	Histogram for the differences between TUB-Shader and TUB-Light using floating point and double precision configuration. . . . .	45

---

5.13	Remaining errors when comparing TUB-Shader and TUB-Light in double precision configuration. . . . .	45
5.14	Topography for the 50 x 50 km analyzed area. . . . .	46
5.15	Comparison of TUB-Light with floating point and double precision configurations. . . . .	46
5.16	Histogram of TUB-Light with floating point and double precision configurations. . . . .	47
6.1	Concept view of the Luna-Resurs lunar lander. . . . .	48
6.2	Topography of the lunar south pole with the candidate Luna-Resurs landing site. . . . .	50
6.3	Topography of an area of 40 x 40 km centered at the Luna-Resurs landing site. . . . .	50
6.4	Accumulated Illumination for an area of 40 x 40 km over a 20-year period at 2 m height. . . . .	52
6.5	Trajectory of the Sun and shape of the horizon. . . . .	52
6.6	Illumination pattern over a one-year period at 2 m height. . . . .	53
7.1	Location of the Shackleton Vicinity. . . . .	55
7.2	Accumulated illumination for SV1 over a 20-year period at 2 m height. . . . .	56
7.3	Topography of SV1 candidate landing site. . . . .	57
7.4	Trajectory of the Sun and shape of the horizon for SV1. . . . .	58
7.5	Illumination pattern over a one-year period at 2 m height. . . . .	59
A.1	Example of the structure of a configuration file structure. . . . .	63
A.2	Example of how the tool is executed. . . . .	64
B.1	Output of the script to check if the selected GPU supports double precision. . . . .	66

# List of Tables

2.1	Successfully past Lunar Lander missions. . . . .	9
2.2	List of future Lunar Lander missions. . . . .	10
4.1	List of main uses for supercomputers over the last decades. . . . .	26
5.1	List of different test to evaluate the performance of TUB-Light . . . . .	33
5.2	List of 10 most illuminated spots for an area of 2 x 2 km over a 20-year period at 2 m height. . . . .	35
5.3	List of the 10 most illuminated spots for an area of 20 x 20 km over a 20-year period at 2 m height. . . . .	37
5.4	List of the most illuminated points at CR1 from Gläser et al. [2014b] and this study. . . . .	38
5.5	List of the 10 most illuminated spots for an area of 20 x 20 km over a one-year period at 2 m height. . . . .	38
5.6	List of the 10 most illuminated spots for an area of 50 x 50 km over a 20-year period and at 2 m height. . . . .	40
5.7	List of the 10 most illuminated spots for TUB-Shader and TUB-Light using floating points. . . . .	42
5.8	List of the 10 most illuminated spots for TUB-Shader and TUB-Light using double precision. . . . .	43
6.1	List of the payload on board the Luna-Resurs lander. . . . .	49
7.1	List of the 10 most illuminated spots for SV1 over a 20-year period at 2 m height. . . . .	56
7.2	List of the most illuminated points in SV1 from previous studies with the one found in this study. . . . .	58
A.1	List of APIs, libraries and software needed. . . . .	62

# Abbreviations

<b>RHU</b>	<b>R</b> adioisotope <b>H</b> eating <b>U</b> nits
<b>OpenCL</b>	<b>O</b> pen <b>C</b> omputing <b>L</b> anguage
<b>GPU</b>	<b>G</b> raphics <b>P</b> rocessing <b>U</b> nit
<b>GPGPU</b>	<b>G</b> eneral- <b>P</b> urpose <b>C</b> omputing on <b>G</b> raphics <b>P</b> rocessing <b>U</b> nits
<b>API</b>	<b>A</b> pplication <b>P</b> rogramming <b>I</b> nterface
<b>CUDA</b>	<b>C</b> omputed <b>U</b> nified <b>D</b> evice <b>A</b> rchitecture
<b>GMT</b>	<b>G</b> eneric <b>M</b> apping <b>T</b> ools
<b>LALT</b>	<b>L</b> aser <b>A</b> LTimeter
<b>LOLA</b>	<b>L</b> unar <b>O</b> rbiter <b>L</b> aser <b>A</b> ltimeter
<b>LRO</b>	<b>L</b> unar <b>R</b> econnaisance <b>O</b> rbiter
<b>DTM</b>	<b>D</b> igital <b>T</b> errain <b>M</b> odel
<b>NAC</b>	<b>N</b> arrow <b>A</b> ngle <b>C</b> amera
<b>JAXA</b>	<b>J</b> apan <b>A</b> erospace <b>e</b> Xploration <b>A</b> gency
<b>NASA</b>	<b>N</b> ational <b>A</b> eronautics and <b>S</b> pace <b>A</b> dministration
<b>ICAT</b>	<b>I</b> llumination and <b>C</b> ommunication <b>A</b> nalysis <b>T</b> ool
<b>CPU</b>	<b>C</b> entral <b>P</b> rocessing <b>U</b> nit
<b>PSA</b>	<b>P</b> ermanent <b>S</b> hadowed <b>A</b> reas
<b>GLXP</b>	<b>G</b> oogle <b>L</b> unar <b>X</b> <b>P</b> rize
<b>RoI</b>	<b>R</b> egion of <b>I</b> nterest

# Chapter 1

## Introduction

### 1.1 Motivation

Lunar exploration is a crucial part of space history. Starting in the 1960's, several mission have landed, orbited and driven on the Moon until the late 1970's, when the lunar exploration race decreased its activities. With the beginning of the new century, the Moon has been again in the top priorities for space exploration. The last missions have opened a new interest in the study of the lunar south pole that, due to its own nature, might harbor water-ice. Furthermore, new mission requirements have created new challenges, being the focus on the use of solar power instead of Radioisotope Heating Units (RHU) one of its main goals (Carpenter et al. [2012]). Therefore, a whole new field of studies in illumination calculation and its behavior at the lunar south pole has been opened.

The calculation of illumination conditions is a very time consuming process. The computation time will vary depending on the extension of the area to be calculated and the period of time to be analyzed. The main focus of this thesis is to reduce this computation time and achieve appropriate results compared with previous studies. As an example, we were able to reduce the computation time from a 24 hours process analyzing a 20 x 20 km region over a one-year period to 25 minutes.

In order to achieve such results, a new parallel computation approach was developed. This new approach uses Open Computing Language (OpenCL) which allows us to utilize the Graphic Processing Unit (GPU) to do arithmetical calculations (Munshi et al.

[2011]). Such process is also known as General-Purpose Computing on Graphics Processing Units (GPGPU) and is a rather recent technique (Ghorpade et al. [2012]).

This technique opens the door to a whole new field of studies on how processes can be parallelized and, in case the problem is designed in a way that allows parallel computing, the best approach to successfully achieve the desired results. By using parallel computation techniques, computation times in illumination studies can be drastically reduced. The following chapters of this work will cover all the processes done to achieve such results and the problems that might arise when dealing with such techniques.

## 1.2 Objectives

This thesis presents a parallel programming approach capable of evaluating illumination conditions at the lunar south pole in a rather reduced computation time. The aim of this work is to introduce the tool and validate it by analyzing the results and compare them to previous studies.

As one of the major goals of the software, besides the reduction of the computation time, is to be cross-platform, the language and methods used to develop it have been chosen very carefully.

There are two main application programming interfaces (APIs) that can be used for parallel programming and GPGPU. The most extended API is the Compute Unified Device Architecture (CUDA), which was created by NVIDIA in 2007. This company is well known for being a graphic cards manufacturer and for its seniority in the GPGPU world. Unfortunately, one of the main drawbacks of CUDA is that it can only run on NVIDIA graphic cards and on selected models (NVIDIA. [Accessed: 17 September 2013]).

For this reason, and to keep the software as portable as possible, OpenCL was chosen. OpenCL is an API that will allow us to apply parallel programming computations in either CPUs and/or GPUs. As OpenCL has been adopted by the majority of the hardware developers (Gaster et al. [2008]) there won't be problems at a hardware level although some changes at software level might be needed (StreamComputing. [Accessed: 17 September 2013]). For the graphical representation, the Generic Mapping



Tools (GMT) language has been used as it is well focused on map representation as well as pure graphic representation. GMT has been developed by the University of Hawaii at Manoa and is an open-source and cross-platform tool.

The thesis is organized as follows. Chapter 2 gives an overview of the Moon, the lunar south pole and different missions to date. Chapter 3 focuses on the data, the different computation methods that have been used and a detailed overview of the method used in this study. Chapter 4 gives an overview of sequential and parallel programming and considers different ways to parallelize the given problem. In chapter 5, we focus on the tool itself, analyzing how it works and the results achieved, allowing us to cross-validate it. Chapter 6 focuses on the first case study - the Russian candidate landing site analysis for the Luna-Resurs mission. Chapter 7, focuses on the second case study - the illumination analysis of the Shakleton Vicinity, a potential landing site. Finally, chapter 8 gives a summary of the work and suggestions for future research.

### 1.3 Related work

The Moon has been a long time the focus of a large variety of studies. The study of its topography is of great importance for lunar exploration missions and lunar scientific investigations. Different observation techniques for the observation of the lunar topography, either earth-based using radar interferometry such as Margot et al. [1999] or satellite based using laser altimeters such as Araki et al. [2009] using the Laser Altimeter (LALT) on-board the lunar orbiter Kaguya (SELENE) or the Lunar Orbiter Laser Altimeter (LOLA) on-board the Lunar Reconnaissance Orbiter (LRO) mission.

Small differences in resolution exist between techniques going from rather low resolutions using radar interferometry (Margot et al. [1999]) to a higher resolutions using LALT (Araki et al. [2009]) or LOLA (Smith et al. [2010]).

Although LOLA offers a very high resolution topography at the lunar south pole, the offsets between the different laser tracks will cause striped Digital Terrain Model (DTM). This issue is solved using the combination of Narrow Angle Camera (NAC) DTMs and LOLA DTMs as stated in Gläser et al. [2014a]. As a result, we will obtain a 20 m/pixel resolution adjusted LOLA DTMs.

Three main studies were performed related with illumination conditions at the lunar polar regions in the last 6 years. The first study was carried out by Noda et al. [2008] from the Japan Aerospace Exploration Agency (JAXA) using data from the Japanese mission Kaguya. For this study the DTM that was used had a resolution of 470 m/pixel for a region from  $-90^\circ$  to  $-85^\circ$ . This DTM was computed using data derived from LALT. The calculation was performed using ray-tracing methods which will be discussed later.

Another investigation was conducted by Bussey et al. [2010] from the Johns Hopkins University, using a DTM derived from LALT and using the same method of computation as Noda et al. [2008]. In this case, the DTM had a resolution of 474 m/pixel for a region from  $-90^\circ$  to  $-85^\circ$ . Although both investigations used almost identical DTMs and the same calculation method, they had different purposes.

While Noda et al. [2008] focused on both poles and used a period of study of 5.5 years, Bussey et al. [2010] centered the investigation only on the south polar region and for a period of one year.

On June 10 2009, the Kaguya spacecraft was programmed to impact on the lunar surface leading to the end of the mission. The same year on June 18, the LRO spacecraft was launched offering us new data.

Mazarico et al. [2011] carried out a new investigation using LOLA DTMs. As LOLA tracks are denser due to the pulse repetition rate and the 5-spot pattern (Smith et al. [2010]), the derived topography had a higher resolution than previous studies. The work is based on a DTM with a resolution of 240 m/pixel for a region from  $-90^\circ$  to  $-80^\circ$ . For the analysis, the horizon method was chosen. The use of a high resolution DTM and the horizon method lead to improved results in less computational time. A discussion of the horizon method will be shown in the following sections.

In the scope of the European Lunar Lander mission (see Carpenter et al. [2012]) a new investigation was done. The investigation in De Rosa et al. [2012] used a new tool developed by EADS-Astrium. The tool is called Illumination and Communication Analysis Tool (ICAT). The DTM used is derived from LOLA data and the horizon method was chosen for the analysis. In this case the DTM had a resolution of just 40 m/pixel for a region from  $-90^\circ$  to  $-80^\circ$ . The main difference of this investigation with respect to the others is the fact that the focus lies on the candidate landing sites for the

ESA Lunar Lander mission instead of a more general view. That is due to the aim of the lander which was projected to use solar arrays instead of RHU. For this reason, a very precise investigation for the illumination was necessary in order to select suitable and safe landing sites.

As stated before, one of the major drawbacks of the ray-tracing and the horizon calculation methods is the computation time. This is caused by how the azimuths and the illumination calculation are programmed. We assume that all the software used up to date are sequentially programmed and use the capacity of the Central Processing Unit (CPU) instead of using the parallel GPU computing performance. While the CPU contains multiple cores (dual-, quad- or octa-core), the GPU involves thousands of cores and, therefore, the performance is increased thus reducing the computation time.

## Chapter 2

# The Moon

### 2.1 Moon overview

The Moon is the closest and the only natural satellite of the Earth and the fifth largest natural satellite in the Solar System. It has a radius of approximately  $1737.4 \text{ km}^1$  and an average distance to the Earth of  $384400 \text{ km}^1$ . The Moon has a sidereal period of 27.3 days<sup>1</sup> (orbital period) and a synodic period of 29.5 days<sup>1</sup>. It is in synchronous rotation with the Earth, meaning that the same face is always turned towards the Earth. The Moon travels at an average speed of  $1.022 \text{ km/s}^1$  around the Earth. The influence of the Earth's gravity and rotation makes the Moon rotational axis change its orientation, resulting in a precessional cycle of 18.6 years<sup>1</sup>. It also has a nutation period of 8.85 years<sup>1</sup> causing a nodding movement in the lunar rotational axis (Figure 2.1).

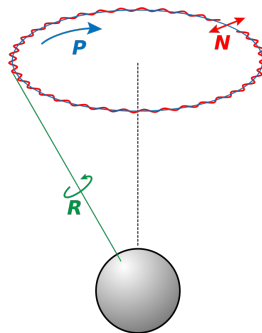


FIGURE 2.1: The rotation (green), precession (blue) and nutation (red) movement of the Moon (Scientific Volume Imaging. [Accessed: 24 october 2014]).

<sup>1</sup>Williams, Dr. David R. "Moon Fact Sheet". NASA (National Space Science Data Center).

The small inclination of its rotational axis ( $1.54^\circ$ ) with respect to the ecliptic plane makes the lunar poles of high interest for the search of water-ice. This combined with the exceptional topography of the lunar south pole, has led the majority of future missions to be focused on this region.

Due to its proximity to the Earth, it has been the only celestial body where humans have physically landed. The first successfully launched mission to the Moon took place in 1959, even though until 1966 no lander made it to the Moon. The first manned landing, Apollo 11, took place in July 1969. Since 1976 with the Luna 24 mission, there has been no lander mission to the Moon with the exception of the Chinese Chang'e 3 which successfully landed in December 2013. New lunar missions such as Luna-Glob, Luna-Resurs, Chang'e 4, Chandrayaan-2 and SELENE-2 are scheduled in the near future.

## 2.2 Lunar south pole

In the last ten years, lunar scientists have been investigating the lunar poles in greater detail. One of the main reasons for focusing on the south pole is the possibility of finding water-ice in permanently shadowed areas (PSA) as well as quasi-permanent illuminated points, where almost continuous illumination is possible. This situation makes possible a long term mission (lunar base or lunar station) or using long illuminated regions to take advantage of the solar power to produce energy for a lander (De Rosa et al. [2012]). Figure 2.2 shows the inclination of the Moon axis with respect to the ecliptic plane that, combined with the exceptional topography of the lunar south pole, makes PSA areas plausible.

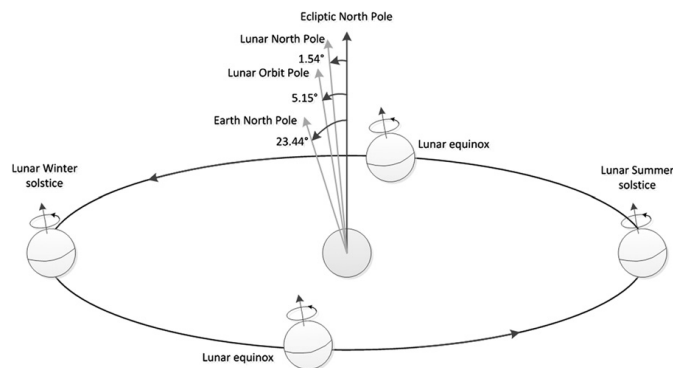


FIGURE 2.2: Inclination of the Moon with respect to the ecliptic plane as well as the lunar equinoxes. (De Rosa et al. [2012])

Due to the LRO mission and the LOLA data set, the scientific community has now access to a high resolution data set for the Moon and specially for the polar regions. This combined with the adjusted DTMs computed in Gläser et al. [2014a], high accuracy results can be expected.

### 2.3 Lunar lander missions

As stated before, the Moon has been a target for orbiters, impactors, landers and rover missions since the mid-twentieth century. Six organizations have successfully launched lunar missions. The former Russian Aviation and Space Agency (ROSAVIKOSMOS), now The Russian Federal Space Agency (ROSCOSMOS), NASA, JAXA, ESA, the Indian Space Research Organization (ISRO) and the China National Space Administration (CNSA).

Table 2.1 shows a chronologically ordered list of relevant and successful lunar lander missions that took place in the past. Note, that besides the Chinese Chang'e 3 mission, we have a gap of 40 years with no landing missions. Thus, combined with the fact that for the scientific community a constant flux of data is necessary, new lunar mission are needed.

As the south pole of the Moon has recently become a target for lunar exploration, mainly for the possibility of finding water-ice, in Table 2.2 a list of future landing missions is shown. One of the major changes in lunar exploration is the introduction of the private sector (Laurini et al. [2010]). The mission aims of such private companies can be divided in two groups, educational and commercial purposes.

In the educational and research group, the leading program is the Google Lunar X Prize (GLXP) space competition aiming to successfully launch a spacecraft capable of landing and traveling across the lunar surface while sending images back to Earth. Commercially, the leading program is Shackleton Energy company, aiming to reach the lunar surface and undertake lunar prospections. The main goal of this company is to use the water-ice reservoirs to establish a network of refueling stations in a low earth orbit or the lunar surface, providing fuel for possible long term missions.

Name	Launch Date	Operator	Mission Type
Luna 9	31 January 1966	Soviet Union	Lander
Surveyor 1	30 May 1966	United States	Lander
Luna 13	21 December 1966	Soviet Union	Lander
Surveyor 3	17 April 1967	United States	Lander
Surveyor 5	8 September 1967	United States	Lander
Surveyor 6	7 November 1967	United States	Lander
Surveyor 7	7 January 1968	United States	Lander
Apollo 11	16 July 1969	United States	Manned Lander
Apollo 12	14 November 1969	United States	Manned Lander
Luna 16	12 September 1970	Soviet Union	Sample Return
Luna 17	10 November 1970	Soviet Union	Lander/Rover
Apollo 14	31 January 1971	United States	Manned Lander
Apollo 15	26 July 1971	United States	Manned Lander/Rover
Luna 20	14 February 1972	Soviet Union	Sample Return
Apollo 16	16 April 1972	United States	Manned Lander/Rover
Apollo 17	7 December 1972	United States	Manned Lander/Rover
Luna 21	8 January 1973	Soviet Union	Lander/Rover
Luna 24	9 August 1976	Soviet Union	Sample Return
Chang'e 3	1 December 2013	China	Lander/Rover

TABLE 2.1: Successfully past Lunar Lander missions. Note that during the 1960's and the 1970's all of the missions were carried out by the Soviet Union and the United States.

Name	Expected Date	Operator	Mission Type
Google Lunar X Prize	2015	Private	Lander/Rover
Chang'e 4	2015	China	Lander/Rover
Luna-Glob 1	2016	Russia	Lander/Rover
Chandrayaan-2	2017	India	Orbiter/Lander/Rover
Chang'e 5	2017	China	Sample Return
SELENE-2	2017	Japan	Lander/Rover
ILN Node 1	2018	United States	Lander
Luna-Glob 2 (Luna-Resurs)	2019	Russia	Lander
Shackleton Energy Company	2019	Private	Lander/Driller
Golden Spike Company	2020+	Private	Lander
OpenLuna	2020+	Private	Lander

TABLE 2.2: List of future Lunar Lander missions. Note the introduction of new countries in the lunar exploration as well as private industries



# Chapter 3

## Methodology

### 3.1 Data

The data is one of the most important parts of illumination studies. All the results obtained in illumination studies rely on how precise the data is and all the derived products (horizons, illumination, slope, etc) depend on the principal data, the DTM. In our study we need a DTM as precise as possible.

Previous studies used data from Kaguya [Noda et al., 2008] and [Bussey et al., 2010] or LRO data [Mazarico et al., 2011] and [De Rosa et al., 2012]. For this investigation, the DTMs used are the ones developed by our institute as commented in Gläser et al. [2014a].

One of the reasons of using these DTMs is that, as they are obtained by adjusting the LOLA tracks to the NAC DTMs, it derives into adjusted LOLA DTMs without visible tracks. This minimizes the data gaps (no outliers), allowing us to obtain better results compared to only using the nominal LOLA tracks. It also allows us to obtain better results than just using NAC DTMs that, as a result of the large shadows in the lunar south pole might contain considerable data gaps. Figure 3.1 shows the improved DTMs in an area located around the potential Connecting Ridge (CR1) landing site near the lunar south pole. In Figure 3.1(a) the NAC DTM derived out of the images from the NAC camera is presented. Figure 3.1(b) shows the LOLA DTM where the laser tracks can be easily detected. As a final result, Figure 3.1(c) shows the final adjusted LOLA

DTM. Note the lack of data gaps compared to Figure 3.1(a) and offset tracks as in Figure 3.1(b).

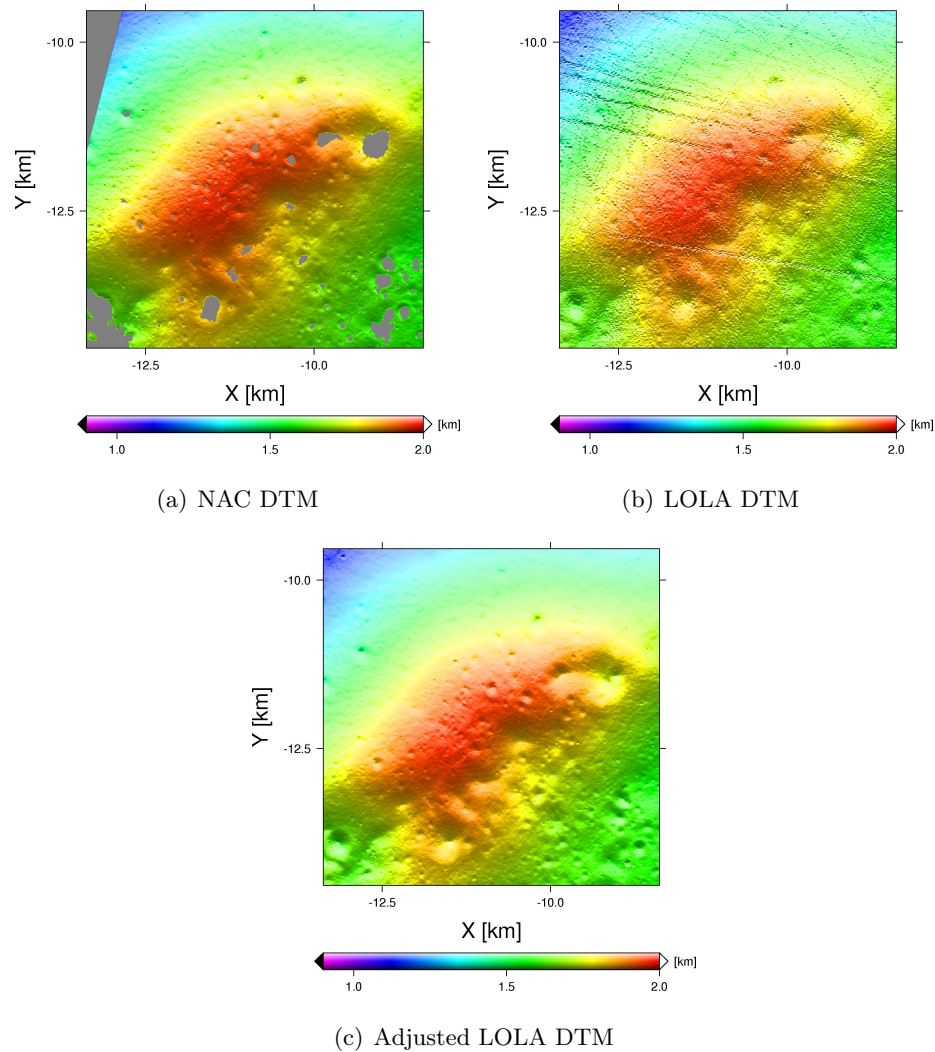


FIGURE 3.1: Comparison of NAC, LOLA and adjusted LOLA DTM. (a) shows the NAC DTM with its data gaps due to the shadows. (b) shows the LOLA DTM, where the slightly offset LOLA tracks can be seen. (c) shows the final adjusted LOLA DTM. (Gläser et al. [2014b])

## 3.2 Computation methods

In illumination studies two main computation methods have been used, the ray-tracing method as in Noda et al. [2008], Bussey et al. [2010] and the horizon method as in Mazarico et al. [2011] and De Rosa et al. [2012]. The methods differ in the calculation process and, as a result of that, the computation time. The results are expected to

concur although performance might vary. Depending on the length of the period or the extension of the region of the study, one or the other method can be applied.

The basic geometrical scenario that has to be faced is presented in Figure 3.2. The position of the Sun in relation to a location on the Moon's surface needs to be computed. The portion of visible Sun is then calculated by analyzing both the elevation of the Sun and the elevation of the highest feature along its direction from the observer.

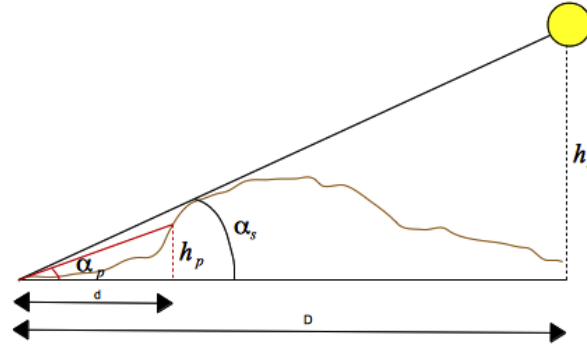


FIGURE 3.2: Basic example of the calculation of the elevation for a terrain feature and for the Sun.  $D$  corresponds to the distance to the Sun and  $d$  to the distance to the terrain feature.  $h_p$  and  $h_s$  correspond to the height of the terrain feature and the Sun respectively, and  $\alpha_p$  and  $\alpha_s$  to the elevation of the terrain feature and the Sun.

A very basic calculation, based on Figure 3.2, can be reduced to the solution of the following equations.

$$\alpha_p = \operatorname{atan}\left(\frac{h_p}{d}\right)$$

$$\alpha_{sun} = \operatorname{atan}\left(\frac{h_s}{D}\right)$$

Although this solution shows a basic overview of how the results will be, several considerations, such as the Moon curvature, have to be contemplated.

The calculation can be divided into following steps: The first step is the comparison of the elevation of the Sun with respect to an observer on the surface of the Moon to the maximum elevation of a terrain feature in the same direction. In a second step, the amount of visible solar-disk as seen by the observer is calculated. While the first step determines whether the Sun is visible or not, the second step calculates the amount of illumination from the Sun.

As the curvature of the Moon can not be neglected, and to reduce the mathematical complexity, vector algebra has been considered. Three main vectors were computed: one vector from the center of the Moon to the location of our observer,  $v_{obs}^{\vec{}}$ , another vector from the center of the Moon to a location on the lunar surface,  $v_{terr}^{\vec{}}$ , and another vector from the center of the Moon to the position of the Sun at a given time,  $v_{m-s}^{\vec{}}$ . All three vectors are given in Cartesian coordinates. (Figure 3.3)

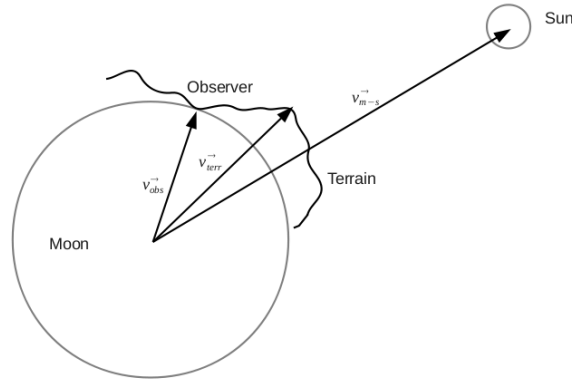


FIGURE 3.3: Vectorial solution scenario with the 3 initial vectors.

The vector from the observer to the horizon,  $v_{hor}^{\vec{}}$ , is calculated by subtracting  $v_{obs}^{\vec{}}$  from  $v_{terr}^{\vec{}}$ . The vector from the observer to the Sun,  $v_{sun}^{\vec{}}$ , is calculated by subtracting  $v_{obs}^{\vec{}}$  from  $v_{m-s}^{\vec{}}$ . The last step is to calculate the angle from each of the two vectors ( $v_{hor}^{\vec{}}$ ,  $v_{sun}^{\vec{}}$ ) to the tangential plane located at the observer location. These elevation angles can now be compared. (Fig. 3.4)

In section 3.2.3 a more detailed explanation of the approach will be shown.

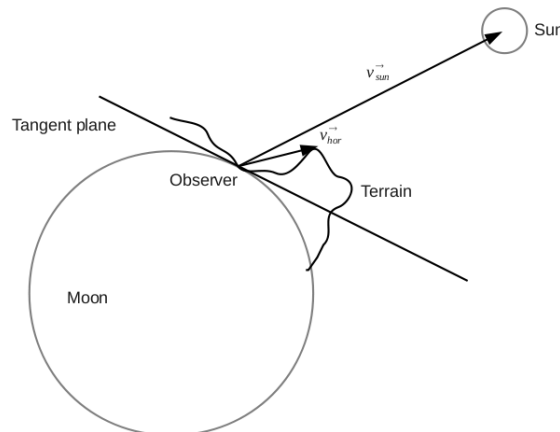


FIGURE 3.4: Vectors involved in the calculation of the elevations.

### 3.2.1 Ray-tracing method

In the studies conducted by Noda et al. [2008] and Bussey et al. [2010] the ray-tracing method was used. The work-flow of the method can be summarized and explained as follows:

1. Load the Moon-Sun geometry data for a specific time.
2. Calculation of the elevation from the Sun to all points of our DTM on the Lunar surface.
3. All points are analyzed to see if the Sun is blocked by a terrain feature along the path
4. Once finished, the next time is loaded and steps 2 and 3 are repeated for all chosen epochs.

The ray-tracing method has strengths and weaknesses. One of the main strength is that the whole elevation computation and visible solar-disk calculation is achieved in one step. However, the main weakness is that the computation time is highly dependent on the size of the region and the length of the period analyzed. This makes the method useful for short periods but very disadvantageous for long study periods.

Due to the nature of our investigation - analyzing large periods of time, the ray-tracing method would be very time consuming. For that reason this method is not pursued further.

### 3.2.2 Horizon method

In recent investigations carried out by Mazarico et al. [2011] and De Rosa et al. [2012] the horizon method is used. The main difference to the ray-tracing method is that the horizon for each pixel in the region of interest (RoI) is calculated and stored as the maximum elevation in different azimuthal directions (e.g. all the maximum elevations for the azimuth  $0^\circ$  are stored in the same file). Once the horizon for each pixel is calculated, the elevation of the Sun with respect to each pixel is computed. This method separates

the whole computation process in two parts: First, the creation of a database with elevations sorted by azimuths and secondly, the computation of illumination.

In our study, the calculation of the azimuth database is done using a tool developed by our institute, TUB-Shader. The parallel programming tool, TUB-Light, focuses on the calculation of illumination using the horizons derived from the first step. A GPU approach for the horizon computation was considered but development was put on hold due to several difficulties during the implementation. We find out that, with the knowledge achieved up to date, there was no possible GPU parallel solution. TUB-Shader is also a parallel tool developed under pthreads that uses the CPU cores to parallelize the computation of the horizons. As this tool was already developed and has been proven to obtain reliable results, no further development of a GPU parallel approach was pursued.

The horizon method can be solved using two different approaches. The main difference between the approaches lies in the order of the iterations to calculate the horizons, being reflected in the computation time.

- First approach:

1. Starting on the first pixel, the computation of all the elevations along the line of sight in  $0^\circ$  azimuth is performed. Only the highest elevation along this line is stored in the first pixel of a file called “0 azimuth”.
2. With a step size of  $0.5^\circ$  all azimuth directions are computed as seen from the first pixel like in step 1. Note, that other than at  $0^\circ, 90^\circ, 180^\circ$  and  $270^\circ$  interpolation is needed to retrieve height information from a gridded DTM. The output will be stored in a new file named with the current azimuthal direction.
3. Steps 1 and 2 are repeated for all the pixels. Note, that with a step size of  $0.5^\circ$  a total of 720 maps will be created.

- Second approach:

1. Starting on the first pixel, the computation of all the elevations along the line of sight in  $0^\circ$  azimuth is performed. Only the highest elevation along this line is stored in the first pixel of a file called “0 azimuth”. Repeat the

same procedure for the next pixel and store the highest elevation in the same 0 azimuth file. The process is repeated until all the highest elevations for azimuth  $0^\circ$  within the selected RoI are calculated.

2. Starting again on the first pixel, the process in step 1 is repeated for azimuth  $0.5^\circ$  and the highest elevation is stored in a file called “0.5 azimuth”. Note, that with a step size of  $0.5^\circ$  a total of 720 maps will be created.

Comparing the performance between the two approaches, the second approach turns out to be faster than the first. The explanation for this behavior is given by the fact that the first approach iterates around pixels and the second iterates around azimuths. Therefore, while the first approach opens and closes each file 720 times multiplied by the number of pixels, the second approach opens and closes each file just once. Since input-output operations are quite slow one should avoid unnecessary file operations. TUB-Shader uses the second approach.

### 3.2.3 Illumination computation

As seen in Figure 3.2, after the elevation of the terrain,  $\alpha_p$  and the elevation of the sun  $\alpha_{sun}$ , are computed they have to be compared to estimate the visibility of the solar-disk. In a very basic approach the decision is made by checking the following conditions:

- Case 1:  $\alpha_{sun} > \alpha_p \rightarrow$  LIGHT
- Case 2:  $\alpha_{sun} < \alpha_p \rightarrow$  NO LIGHT

Since the Sun is treated as an extended source rather than a point source, a more complex approach is adopted. The illumination is evaluated considering a diameter of  $0.533^\circ$  for the solar-disk, which is the apparent size in degrees of the Sun as seen from the Moon.

The illumination computation is summarized in two steps: the calculation of the visible portion of solar-disk (Figure 3.5) and the assignation of a numerical value based on the previous calculation.

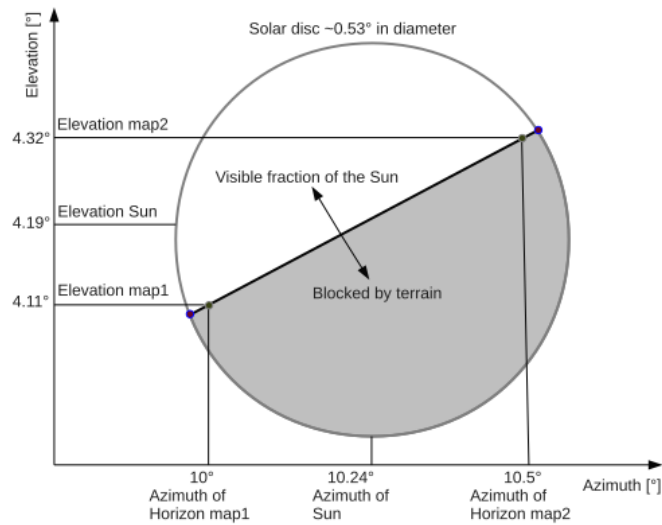


FIGURE 3.5: Calculation of the amount of visible Sun from a certain location (Gläser et al. [2014b]).

First, the elevation of the horizon for the corresponding Sun azimuth was computed. This is done using the following linear interpolation equation.

$$\alpha_{hz} = \alpha_{p_{min}} + (\alpha_{p_{max}} - \alpha_{p_{min}}) \cdot \frac{\theta_{sun} - \theta_{min}}{\theta_{max} - \theta_{min}}$$

The equation considers the maximum elevation of the preceding azimuth  $\alpha_{p_{min}}$ , and the succeeding azimuth  $\alpha_{p_{max}}$ , the Sun azimuth  $\theta_{sun}$ , and the azimuths of the preceding and the succeeding elevations,  $\theta_{min}$  and  $\theta_{max}$  respectively.

Secondly, the position of our observer in gnomonic projection was reduced to Cartesian coordinates. The reduction is done using the formulas in Snyder [1987]. Note that all values must be in radians.



$$\begin{aligned}
\rho &= \sqrt{x_{obs}^2 + y_{obs}^2} \\
c &= \arctan\left(\frac{\rho}{r_{moon}}\right) \\
\lambda &= \lambda_0 + \arctan\left(\frac{x_{obs} \cdot \sin(c)}{\rho \cdot \cos(\phi_0) \cdot \cos(c) - y_{obs} \cdot \sin(\phi_0) \cdot \sin(c)}\right) \\
\phi &= \arcsin\left(\cos(c) \cdot \sin(\phi_0) + \left(\frac{y_{obs} \cdot \sin(c) \cdot \cos(\phi_0)}{\rho}\right)\right) \\
x_{cart} &= (r_{moon} + z_{obs}) \cdot \cos(\phi) \cdot \cos(\lambda) \\
y_{cart} &= (r_{moon} + z_{obs}) \cdot \cos(\phi) \cdot \sin(\lambda) \\
z_{cart} &= (r_{moon} + z_{obs}) \cdot \sin(\phi)
\end{aligned}$$

$x_{cart}$ ,  $y_{cart}$  and  $z_{cart}$  are the Cartesian coordinates of the observer. These coordinates are the components of the vector from the center of the Moon to the observer,  $v_{obs}^{\vec{}}$ .

In the next step, the vector from the center of the Moon to the location of the highest terrain feature,  $v_{terr}^{\vec{}}$ , and the vector from the center of the Moon to the position of the Sun at a given time,  $v_{m-s}^{\vec{}}$ , were calculated.

In a final step, the angle from the observer to the horizon,  $\alpha_{hor}$ , and from the observer to the Sun,  $\alpha_{sun}$ , were calculated (Figure 3.6). This is achieved taking advantage of vector algebra and using the arccosine of the dot product.

$$\begin{aligned}
\alpha_{sun} &= 90^\circ - \arccos \frac{v_{obs}^{\vec{}} \cdot v_{m-s}^{\vec{}}}{\|v_{obs}^{\vec{}}\| \cdot \|v_{m-s}^{\vec{}}\|} \\
\alpha_{hor} &= 90^\circ - \arccos \frac{v_{obs}^{\vec{}} \cdot v_{terr}^{\vec{}}}{\|v_{obs}^{\vec{}}\| \cdot \|v_{terr}^{\vec{}}\|}
\end{aligned}$$

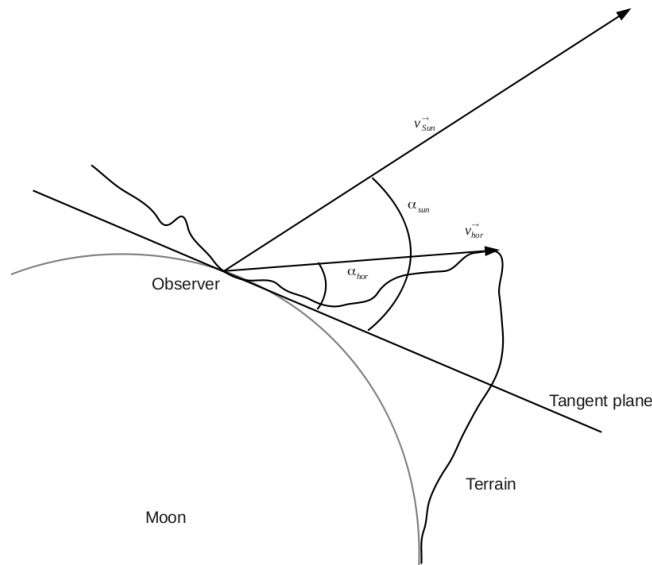


FIGURE 3.6: Calculation of the elevations using vector algebra.

As the Sun is considered a solar-disk, the intersection of the Sun with the horizon was computed.

The mathematical approximation of such situation is the so-called circular segment problem. With this approximation, the intersection of the Sun with the horizon at one particular azimuth and the amount of visible solar-disk were computed.

First, the equation of a line and the equation of a circle are combined. The result of the combination is a quadratic equation.

$$y = mx + c$$

$$(x - x_c)^2 + (y - y_c)^2 - r_{sun}^2 = 0$$

Combining the two equations results in the following equation.

$$x^2 \cdot (1 + m^2) - 2 \cdot x \cdot (x_c - m \cdot c + y_c \cdot m) + (x_c^2 + y_c^2 + c^2 - 2 \cdot y_c \cdot c - r_{sun}^2) = 0$$

The equation is rearranged in three parameters A, B and C.

$$\begin{aligned}
 A &= 1 + m^2 \\
 B &= -2 \cdot x_c + 2 \cdot m \cdot c - 2 \cdot y_c \cdot m \\
 C &= x_c^2 + y_c^2 + c^2 - 2 \cdot y_c \cdot c - r_{sun}^2
 \end{aligned}$$

Secondly, the possible line-circle intersection was analyzed. This was done by calculating the so-called discriminant ( $\Delta$ ).

$$\Delta = B^2 - 4 \cdot A \cdot C$$

The discriminant of a polynomial is a function that gives information about the nature of its roots. In this case, the discriminant has 3 possible solutions,  $\Delta > 0$  meaning that there are two real roots,  $\Delta = 0$  meaning that one real double root exist and  $\Delta < 0$  meaning that the polynomial has no real roots. In this case the line will intersect if and only if  $\Delta > 0$ .

To assign a value for the non-intersection situations, the position of the Sun with respect to the horizon line was analyzed. In case the Sun is above the horizon, a value of 1 is assigned and if it lies below the horizon a value of 0 is assigned (Figure 3.7).

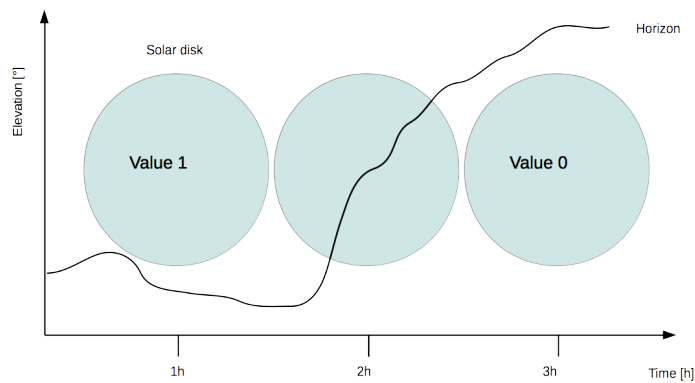


FIGURE 3.7: Value assignment for the non-intersecting cases.

If the line intersects with the circle, the quadratic equation and the line equation are solved to obtain the two intersection points.

$$x_1 = \frac{-B - \sqrt{\Delta}}{2 \cdot A}$$

$$y_1 = x_1 \cdot m + c$$

$$x_2 = \frac{-B + \sqrt{\Delta}}{2 \cdot A}$$

$$y_2 = x_2 \cdot m + c$$

Considering Figure 3.8 as an example, the visible circular segment was calculated. Note that Figure 3.8 reflects the case that less than the 50% of the Sun is visible.

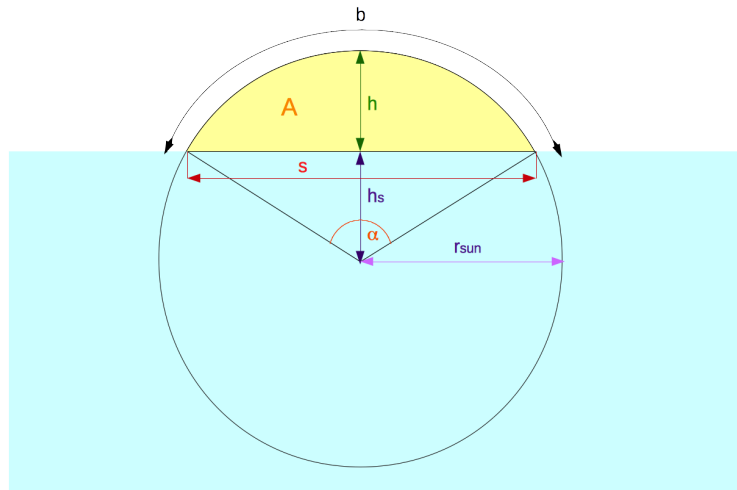


FIGURE 3.8: Example of calculation of the visible circular segment. Note that this figure reflects the case that less than the 50% of the Sun is visible.

$$h = r_{sun} - \sqrt{\left(x_c - \frac{1}{2} \cdot (x_1 + x_2)\right)^2 + \left(y_c - \frac{1}{2} \cdot (y_1 + y_2)\right)^2}$$

$$h_s = r_{sun} - h$$

$$s = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

$$\alpha = 4 \cdot \arctan \frac{2 \cdot h}{s}$$

$$b = r_{sun} \cdot \alpha$$

$$A = r_{sun} \cdot \frac{b}{2} - s \cdot \frac{r_{sun} - h}{2}$$

In a final step, the portion of the circular segment was analyzed to differentiate between the visible portion or the shadowed area. This is due to the fact that this process calculates the circular segment for  $\alpha \leq 180$ . If the center of the Sun is above the horizon, the obtained result is the non-visible area instead of the visible area (Figure 3.9).

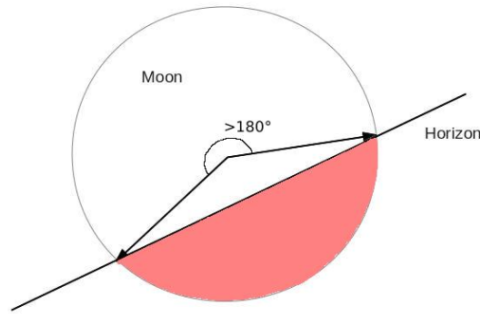


FIGURE 3.9: Example of a wrong calculation of the amount of visible solar disc. The calculated area is red area laying below the horizon when the correct visible area is the white area.

To calculate the correct visible area, the position of the center of the Sun was analyzed and assigned a value between 0 and 1.

$$visible = \begin{cases} \frac{1-A}{\pi \cdot r_{sun}^2} & \text{if } \alpha_{sun} > \alpha_{hor} \\ \frac{1-(\pi \cdot r_{sun} - A)}{\pi \cdot r_{sun}^2} & \text{if } \alpha_{sun} \leq \alpha_{hor} \end{cases}$$

This process is repeated for all the time steps in the evaluated period. The results are stored in a file with illumination values for all the pixels in the RoI. In our study simulations over a period of 1 year and 20 years for different RoIs were conducted. The results are commented in Chapter 5.

## Chapter 4

# Programming techniques

### 4.1 Sequential programming

Sequential programming can be defined as a consecutive and ordered way to execute processes one after another. Thus being modeled after a chronological sequence of events. It is a way to execute processes always finishing the previous before launching the next one as in Figure 4.1

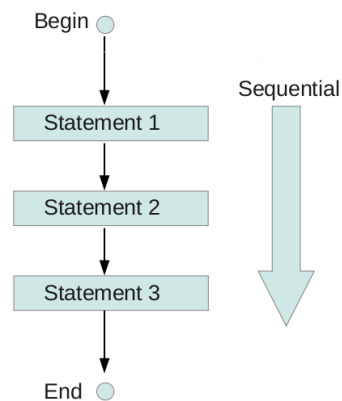


FIGURE 4.1: Graphical example of sequential programming work-flow. A process is not launched before the previous is finished.

This programming technique is used in the most important languages such as C, C++, Pascal, FORTRAN, Python and Java.

In sequential programming the processing unit where the code is executed is typically the CPU. Depending on the clock speed of the processor the program will run faster or slower.

While in sequential programming the program runs in one platform and in a delimited number of cores, in parallel programming we can use more than one platform (heterogeneous) and all the cores of these platforms (multi-core) to run our program. This allows us to execute processes concurrently reducing the computation time.

## 4.2 Parallel programming

Parallel computing is a technique that uses multiple processing platforms simultaneously to solve a problem. In its origins, parallel computing was exclusively dedicated to large supercomputers. These computers are built in very large facilities due to their size and energy consumption, and can execute up to thousands processors running in parallel (Figure 4.2).



FIGURE 4.2: Example of supercomputer. In this case the model is an IBM Blue Gene/P. Note the big size of the components and the large facility. (Argonne National Laboratory. [Accessed: 05 november 2014])

These computers are mainly used for scientific and engineering research. They can be divided in the main trends of research over the last decades in Table 4.1.

---

1970s	Weather forecasting and aerodynamic research
1980s	Probabilistic analysis and radiation shielding modeling
1990s	Brute force code breaking
2000s	3D nuclear test simulations
2010s	Molecular Dynamics Simulation

---

TABLE 4.1: List of main uses for supercomputers over the last decades.

The major problems of such supercomputers are the large size and the high cost of construction and maintenance, which make them very unique computers. The very exclusive access to these computers lead to the creation of new parallel computing techniques, focusing on the parallelization of processes using personal computers.

In order to execute the programs using personal computers, several parallel computing languages were developed. Focusing on GPU languages, the most important are CUDA and OpenCL. To parallelize code on the CPU several multi-threading APIs exist. Pthreads is the most used and is a binding of C and C++ that creates different threads allowing to assign them to each of the cores of the CPU. This language is discarded since it can only run on Linux.

CUDA bases its parallelization in the use of the cores contained in a GPU. It is widely used in computer graphics and video games design. Due to the growth of the use of parallel programming for scientific purposes, CUDA started to focus on this field. Although its use is very extent, it can only run on NVIDIA graphic cards.

For this reason we decided to use OpenCL. It is a cross-platform (Linux, Mac OS and Windows) and heterogeneous (GPU and CPU) language that fulfills all our needs. The language is described in detail in section 4.2.3.

Parallel computing can be divided in two approaches, data parallel and task parallel.

#### 4.2.1 Task parallelism

Task parallelism is a way to divide a problem in different processes. The horizon computation, which comprises 360 azimuths, can be divided in four different processes running 90 azimuths each. Furthermore, 4 cores of our CPU can be used to run such a program.



The calculation of the azimuths between 0 and 89, 90 and 179, 180 and 269, 270 and 359 will run in four different cores at the same time. This will reduce the computation time in an order of 4 to calculate such a problem (Figure 4.3).

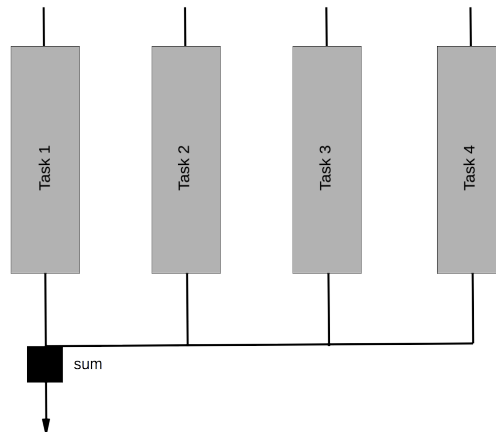


FIGURE 4.3: Graphical example of task parallelism.

Task parallelism is used when the analyzed data is highly dependent on each other.

### 4.2.2 Data Parallelism

Apart from task parallelization, also data can be parallelized. Data parallelism is a completely different approach compare to task parallelism. While task parallelism means running different tasks in parallel using different data, data parallelism means running the same task using the same data in parallel. A very simple example is the vector addition computation. While in sequential programming an entry in vector A is summed up to an entry in vector B and then moved to next entry, in data parallelism all the entries are summed up at the same time (Figure 4.4).

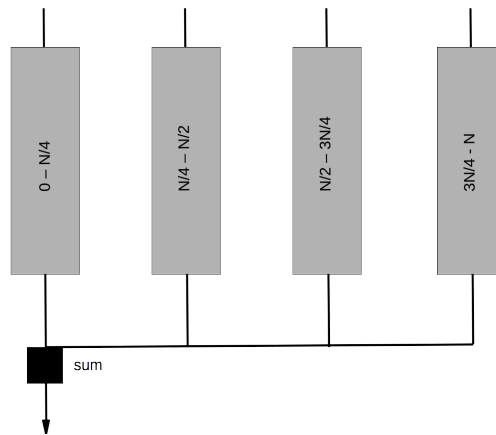


FIGURE 4.4: Graphical example of data parallelism.

As stated before, different languages can be used to parallelize code. In our investigation OpenCL is chosen since it is a cross-platform and open-source language.

### 4.2.3 Introduction to OpenCL

OpenCL is a framework designed by Apple Inc. and developed and maintained by the non-profit consortium Khronos Group. It has been adopted by the vast majority of graphic cards vendors. Although it is a rather young language, since it appeared in late August 2009, it is very powerful and stable.

OpenCL allows to execute programs across different platforms such as CPUs and GPUs (Gaster et al. [2008]). OpenCL is a language based on C, allowing to parallelize both - tasks and data. This makes it a very broad and versatile language.

For this investigation, OpenCL is chosen since it is a cross-platform and heterogeneous language, and it is mainly focused on GPGPU applications.

OpenCL allows us to run and test TUB-Light in both the CPU and the GPU, permitting to compare the performance of the tool running in different platforms. This was also one of the key reasons to choose OpenCL.

#### 4.2.4 CPU vs. GPU

Parallel computing can be executed on different platforms. In this investigation we centered the efforts in running the tool on the GPU. Comparing the cores contained in a CPU and a GPU, it can be noticed that while the CPU can contain up to 8 cores, the GPU contains thousands of computing cores in one single machine (Gupta and Babu [2011]) (Figure 4.5).

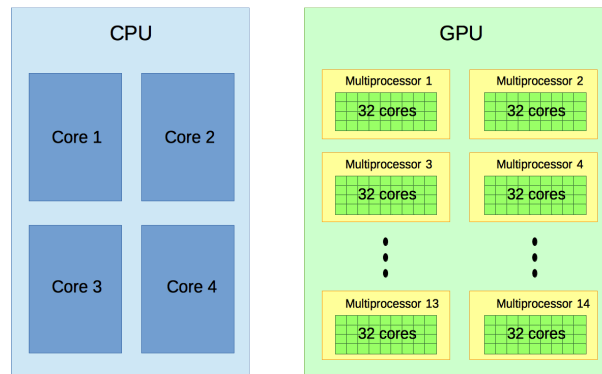


FIGURE 4.5: Comparison of the number of cores contained in a CPU to a GPU.

Using parallel programming on the CPU already increases the performance of a program however, the highest performances are achieved when the program is run on the GPU. Figure 4.6 shows the computation time versus the amount of input data for a vector addition problem (Marco Figuera et al. [2014]), using sequential and parallel programming on the CPU and parallel programming on the GPU.

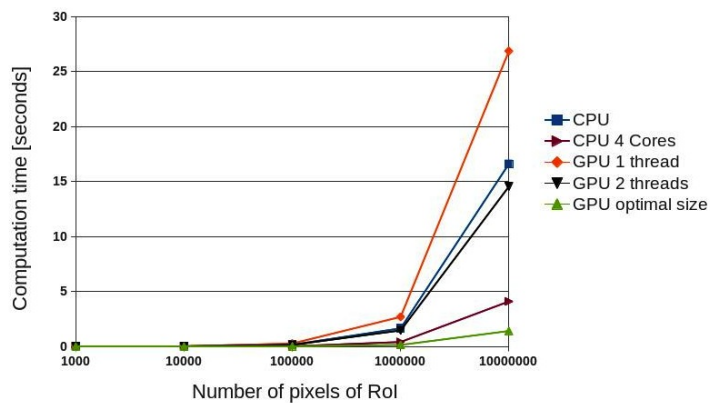


FIGURE 4.6: CPU vs GPU vector addition computation time depending on the amount of input data.

If the code runs on the CPU, one should choose the number of active cores (task parallel). On the GPU the parallelization is performed by establishing the size of the different threads. In other words, the amount of data simultaneously processed (data parallel). Figure 4.6 shows a reduction of the computation time in the order of 3 using 4 cores on the CPU compared to using sequential programming on the CPU.

For the GPU approach, the thread size should be chosen wisely. Figure 4.6 shows an increase of double the time using the GPU approach with a thread size of one (one data entry is sent at a time), compared to the sequential CPU approach. This can be explained by the time needed to send the data to the GPU and read it back. Therefore, the optimal selection of the thread size is of high importance.

The thread size is constrained by the underlying platform. The maximum number of threads to be run concurrently is known by calling an OpenCL built-in function. The most important restriction is that it has to be a multiple of the total amount of input data (RoI in our case) and can not be larger than the maximum number of threads allowed by the platform. Different methods for the automatic selection of the right thread size are under study, although they only work on the CPU (Seo et al. [2013]). Therefore, a manual selection of the thread size is highly recommended. The selection of the thread size is based on the outcome of the built-in function, which will help to set up the limit of the threads considering the input data.

In our investigation, the primary platform to run the simulations is the GPU, selecting the CPU when the simulation can not be executable on the GPU.

## Chapter 5

# TUB-Light a Parallel programming illumination tool

### 5.1 Overview

During the last decades the study of illumination conditions in planetary bodies has arisen. Several planetary bodies like Mars, Mercury or the Moon, have been analyzed. Lunar illumination has focused a large amount of studies and in particular illumination studies in its south pole. As most of the new lunar missions are focused on finding water-ice at the lunar south pole, the study of illumination conditions are of great importance. These studies will help during the mission planning phase.

With the preparation of the new ESA Lunar Lander (Carpenter et al. [2012]), a new illumination study was carried out by De Rosa et al. [2012]. The main focus of this study is to find the most illuminated locations at the lunar south pole. This will help feeding the solar arrays of a lander for heating and power purposes instead of using a RHU.

Several tools for the calculation of illumination conditions were developed using non-parallel implementations. As the computation time using these tools is very time consuming, it would be a great improvement to have parallel tools.

TUB-Light is a parallel computing tool capable of evaluating illumination conditions at the lunar south pole. The tool is programmed using OpenCL and can run on the GPU

and the CPU. With this tool, the computation time has been reduced from 25 hours to 24 minutes for an area of 20 x 20 km over a period of one year. This is a reduction in computational time of factor 62.5.

## 5.2 Motivation

As stated before, the calculation of illumination conditions is a very time consuming process. The computation time depends principally on the combination of the analyzed area, the resolution of the DTM, the period and the time step.

All the tools tested in this work use sequential programming or distributed programming, the high computational time being their main drawback. The reduction of the computation time implies the development of a new tool using parallel programming. This tool can run on the GPU and the CPU of a single computer, allowing us to achieve the same results as previous studies in a reduced time.

## 5.3 Data

The data used in TUB-Light is a set of horizons. Each horizon consist of 720 files containing the maximum terrain elevation as seen from each pixel in our RoI every  $0.5^\circ$ .

All the data was computed using DTMs in gnomonic projection. The use of gnomonic projection allows to calculate the azimuths without having to apply any corrections, since straight lines in the map correspond to great circles on the sphere (Figure 5.1).

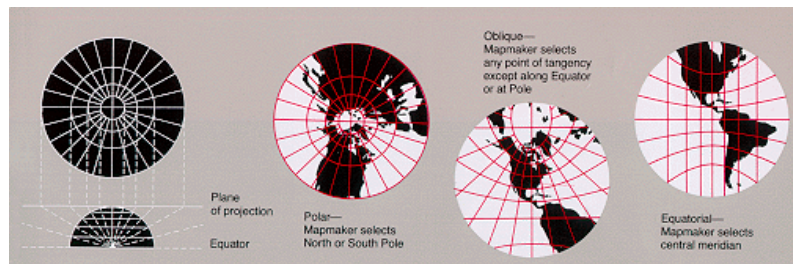


FIGURE 5.1: Gnomonic projection. The oblique case shows the straight line from a starting point as straight lines (USGS. [Accessed: 24 october 2014]).

## 5.4 Results

A set of different scenarios were proposed in order to evaluate the performance of the tool (Table 5.1). All the horizons were computed using a 20 m/pix DTM at 2 m height above the ground.

Test	RoI	Num. Pixels	Period
1	2 km x 2 km	10000	1 year
2	2 km x 2 km	10000	20 years
3	20 km x 20 km	1000000	1 year
4	20 km x 20 km	1000000	20 years
5	50 km x 50 km	6250000	1 year
6	50 km x 50 km	6250000	20 years

TABLE 5.1: List of different test to evaluate the performance of TUB-Light.

In our study, all the tests were run using a Nvidia QuadroFX 1800 GPU with the optimal number of threads, allowing to achieve the best performance of the tool. Table 5.2 shows the computation time needed to calculate the tests. For test 1 and 2 the computation time is 25 seconds and 2.7 minutes respectively, test 3 and 4 take 38 min and 150 min respectively and test 5 and 6 take 236 minutes and 896 minutes.

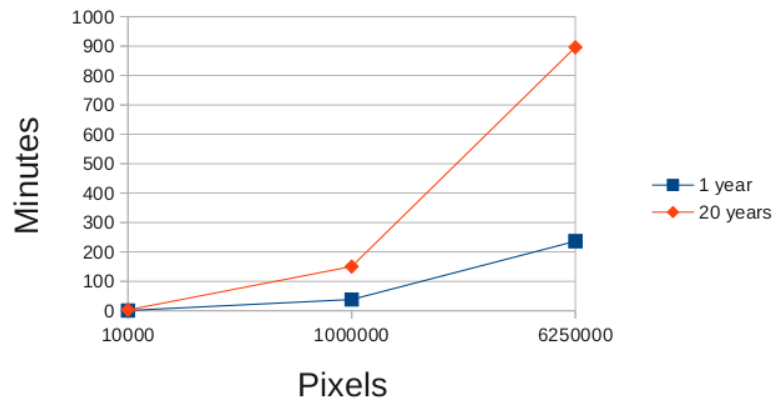


FIGURE 5.2: Computation time for all the tests.

As a result, the accumulated illumination and the illumination pattern are presented for all the tests.

### 5.4.1 2 x 2 km results

In these tests we focused on an area of 2 x 2 km centered at the Connecting Ridge (CR1) potential landing site. In Figure 5.3 the accumulated illumination over a one-year period is presented. Note that the most illuminated spots receive more than 300 days of accumulated illumination, reaching a maximum of 316.875 days.

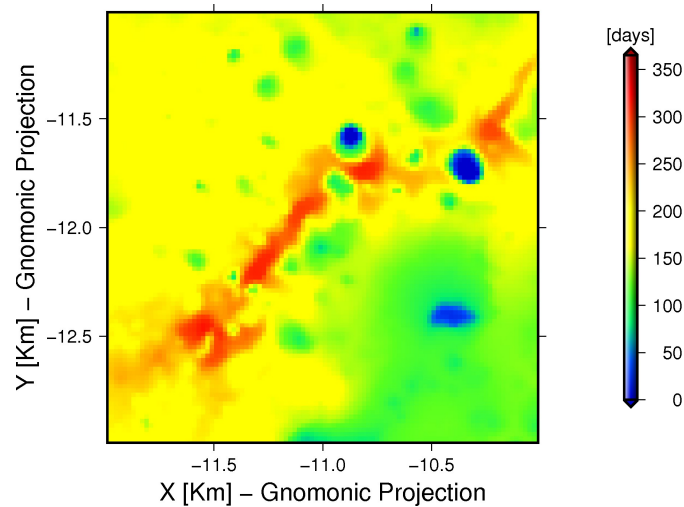


FIGURE 5.3: Accumulated Illumination for an area of 2 x 2 km centered at CR1 over a one-year period at 2 m height.

This highly accumulated illumination can be explained analyzing the Sun pattern in Figure 5.4, where the Sun is visible almost during the whole period. The longest period of constant darkness is 4.58 days and the longest constant illumination of 231.833 days. The extended illumination is due to the location of the spot which resides in higher elevation.

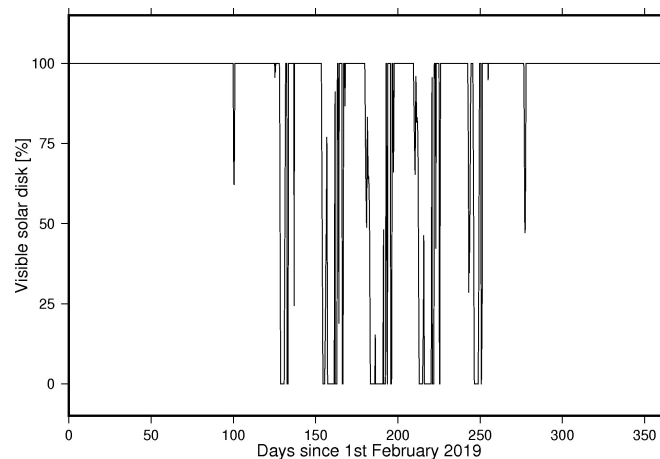


FIGURE 5.4: Illumination pattern located at 89.4394°S, 222.8065°E over a one-year period at 2 m height.



Figure 5.5 shows the Sun pattern over a 20-year period. Analyzing the Sun pattern over 20 year covers the whole precessional cycle (18.6 years), allowing us to study the maximum amount of illumination a spot can receive. This study is very useful in case of a long term mission.

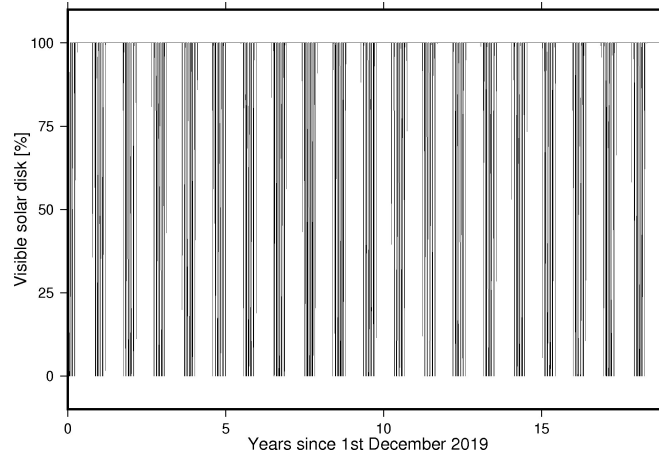


FIGURE 5.5: Illumination pattern located at  $89.4394^{\circ}\text{S}$ ,  $222.8065^{\circ}\text{E}$  for a 20 year period.

In Table 5.2, the 10 most illuminated spots for an area of  $2 \times 2$  km over a 20-year period at 2 m height are presented.

Latitude	Longitude	Accumulated illumination %
-89.4394	222.8065	88.0819
-89.4399	222.8524	87.9122
-89.4404	222.8029	87.2136
-89.4390	222.8559	86.6155
-89.4399	222.7570	86.4732
-89.4739	222.6626	86.2312
-89.4404	222.8983	86.2128
-89.4395	222.9018	86.2128
-89.4385	222.8102	86.1697
-89.4735	222.7154	86.1393

TABLE 5.2: List of 10 most illuminated spots for an area of  $2 \times 2$  km over a 20-year period at 2 m height.

To conclude, Figure 5.6 shows the horizon and trajectory of the Sun at  $89.4394^{\circ}\text{S}$ ,  $222.8065^{\circ}\text{E}$  over a 20-years period. The black lines correspond to the trajectory of the

Sun while the horizon is color-coded with respect to the distance to the highest feature in the terrain. Figure 5.6 reveals that the Sun is almost always above the horizon - illuminating the area for long periods.

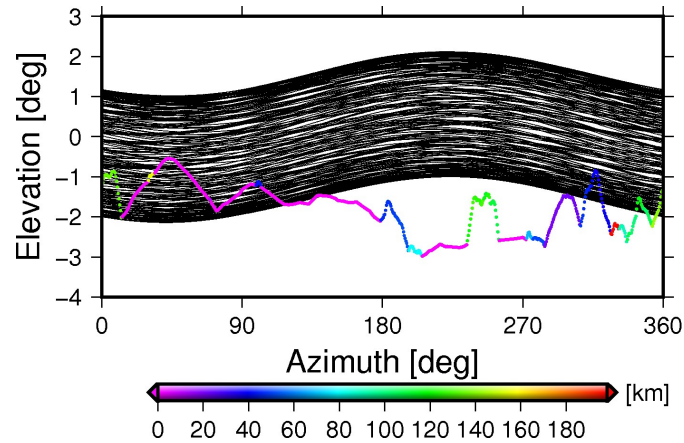


FIGURE 5.6: Trajectory of the Sun and shape of the horizon as seen from  $89.4394^{\circ}\text{S}$ ,  $222.8065^{\circ}\text{E}$ . Black correspond to the path of the Sun over 20 years. The horizon color-codes the distance to the highest elevated feature.

#### 5.4.2 20 x 20 km results

These tests cover an area of 20 x 20 km around the CR1 potential landing site. In Figure 5.7 the accumulated illumination for a 20 x 20 km area at the lunar south pole over a one-year period is presented. Note the large number of highly illuminated areas situated at the CR1 potential landing site and Shackleton area. These areas receive more than 250 days of accumulated illumination.

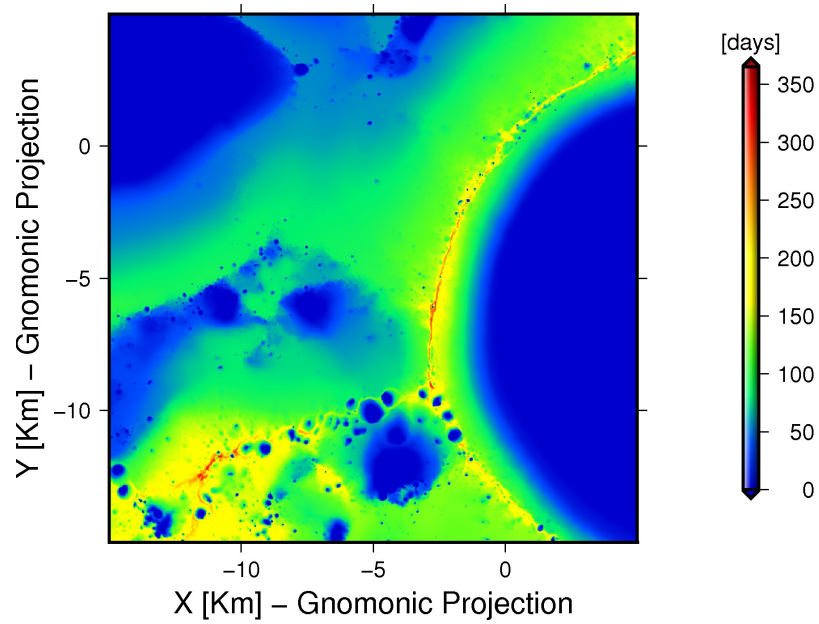


FIGURE 5.7: Accumulated illumination for an area of 20 x 20 km at the lunar south pole over a one-year period at 2 m height.

Table 5.3 shows the 10 most illuminated spots. Note that all the spots are located at the CR1 area.

Latitude	Longitude	Accumulated illumination %
-89.4394	222.8065	88.1103
-89.4399	222.8524	87.9421
-89.4404	222.8029	87.2415
-89.4390	222.8559	86.6293
-89.4399	222.7570	86.4913
-89.4739	222.6626	86.2697
-89.4404	222.8983	86.2090
-89.4395	222.9018	86.2025
-89.4385	222.8102	86.1850
-89.4735	222.7154	86.1741

TABLE 5.3: List of the 10 most illuminated spots for an area of 20 x 20 km over a 20-year period at 2 m height. Marked in red a location varying its position when illumination is calculated over a one-year period (Table 5.5)

The results can be cross-validated with Gläser et al. [2014b] (see Table 5.4).

Study	Res.[m/pix]	Illu.[%]	Location latitude, longitude	h [m]	Period[y]
Gläser et al. [2014b]	20	88.1	-89.4394, -222.8065	2	20
Gläser et al. [2014b]	20	87.9	-89.4399, -222.8524	2	20
This study	20	88.1103	-89.4394, -222.8065	2	20
This study	20	87.9421	-89.4399, -222.8524	2	20

TABLE 5.4: List of the most illuminated points at CR1 from Gläser et al. [2014b] and this study.

Table 5.5 shows the 10 most illuminated spots over a one-year period at 2 m height. Note how the red marked entry in Table 5.3 has moved to a lower position in Table 5.5. This is due to the coverage of the precessional cycle, allowing to quantify the total amount of illumination that a spot can receive.

Latitude	Longitude	Accumulated illumination %
-89.4394	222.8065	88.6445
-89.4399	222.8524	88.5094
-89.4404	222.8029	87.7643
-89.4390	222.8559	87.2288
-89.4399	222.7570	87.0071
-89.4404	222.8983	86.8208
-89.4395	222.9018	86.8094
-89.4385	222.8102	86.7333
-89.4739	222.6626	86.6500
-89.4735	222.7154	86.6142

TABLE 5.5: List of the 10 most illuminated spots for an area of 20 x 20 km over a one-year period at 2 m height. Marked in red is the same location as in Table 5.3 moved to a lower position due to the analyzed period.

Since the most illuminated spot and the period analyzed is the same as in section 5.4.1, the illumination pattern and the trajectory of the Sun and shape of the horizon are not included in this section.

### 5.4.3 50 x 50 km results

In these tests the largest area studied by our department at such resolution is presented. The area covers an extension of 50 x 50 km of the lunar south pole. It covers the complete Shackleton crater including the Shackleton rim potential landing sites (SR1, SR2 and SR3) as well as the potential landing site Shackleton Vicinity (SV1). Figure 5.8 shows the accumulated illumination for the analyzed area over a one-year period at 2 m height.

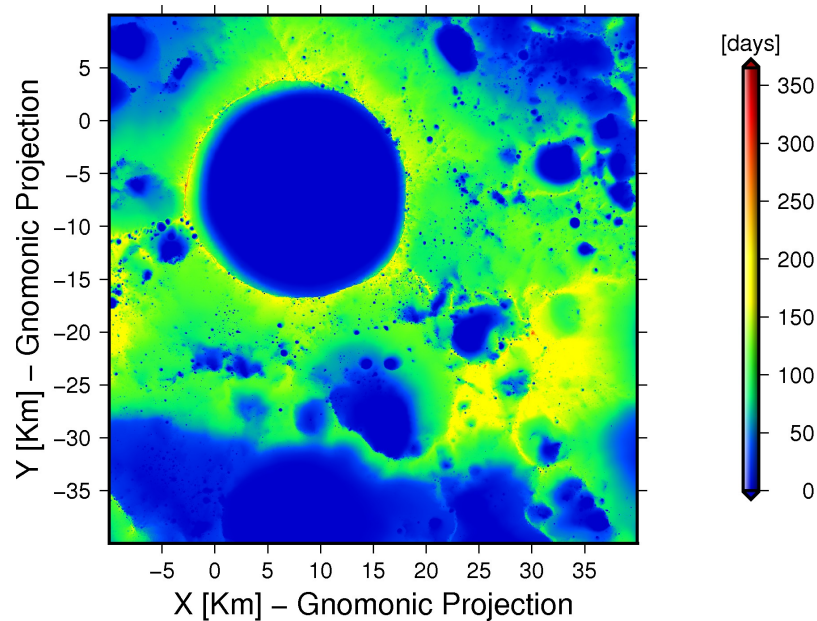


FIGURE 5.8: Accumulated Illumination for an area of 50 x 50 km at the lunar south pole over a one-year period at 2 m height.

The most illuminated locations correspond to SR1, SR2, SR3 and SV1. In Table 5.6 the 10 most illuminated spots in the area are shown. The most illuminated spot corresponds to SR1 and receives an accumulated illumination of 85.5108%.

Latitude	Longitude	Accumulated illumination %
-89.6868	197.2136	85.5108
-89.6873	197.3648	85.2971
-89.6866	197.3288	85.2826
-89.6874	197.2494	85.2653
-89.6862	197.1779	85.0163
-89.6879	197.4009	84.8862
-88.8092	123.6856	84.7740
-88.8088	123.7120	84.5439
-88.8083	123.6944	84.2374
-89.6885	197.4372	84.2022

TABLE 5.6: List of the 10 most illuminated spots for an area of 50 x 50 km over a 20-year period and at 2 m height.

Figure 5.9 shows the trajectory of the Sun for an observer located at  $-89.6868^{\circ}\text{S}$ ,  $197.2136^{\circ}\text{E}$ . The black line corresponds to the trajectory of the Sun while the horizon is color-coded to the distance of the highest elevated feature. Figure 5.10 shows the pattern of the Sun vs. time. The longest period of constant darkness is 2.7 days and the longest period of constant illumination is 83.08 days. The analyzed location for the selected period shows that the Sun is above the horizon most of the times.

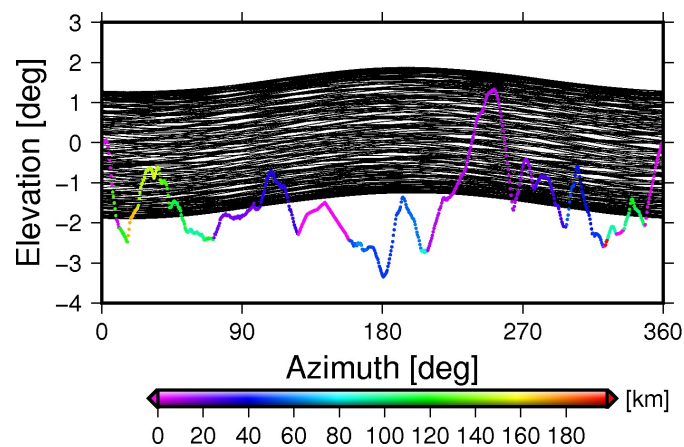


FIGURE 5.9: Trajectory of the Sun and shape of the horizon as seen from  $-89.6868^{\circ}\text{S}$ ,  $197.2136^{\circ}\text{E}$ . Black corresponds to the path of the Sun. The horizon is color-coded to the distance of the highest elevated feature.

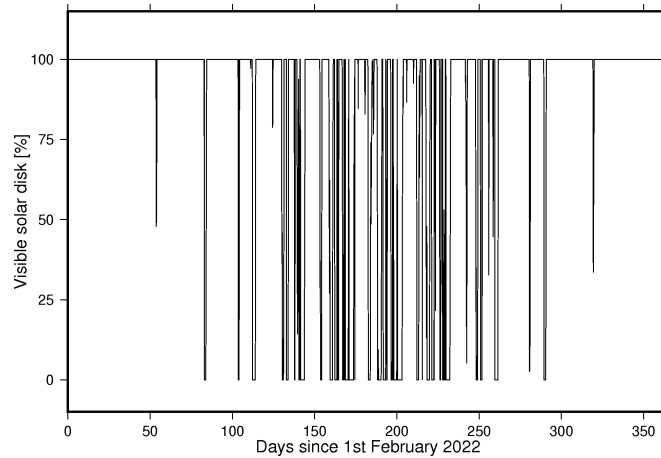


FIGURE 5.10: Illumination pattern for a location at  $-89.6868^{\circ}\text{S}$ ,  $197.2136^{\circ}\text{E}$  over a one-year period at 2 m height.

#### 5.4.4 Cross-validation with TUB-Shader

In order to cross-validate the results obtained with TUB-Light, we compared our results with TUB-Shader. Using the results from TUB-Shader allows us to validate our results with a tool that uses the same input data as TUB-Light.

TUB-Shader is a tool developed by our institute capable of calculating illumination conditions. This tool also uses the DTMs developed by our institute. It is parallelized using pthreads which is a Portable Operating System Interface (POSIX) for threads. This allows TUB-Shader to distribute the data work load on multiple CPU cores. In this case, it creates 4 different threads (corresponding to 4 cores) working in parallel.

For the cross-validation test the 10 most illuminated spots for an area of 50 x 50 km over a 20-years and at 2 m height were compared. The area analyzed is shown in Figure 5.8.

Table 5.7 shows the results of the cross-validation. The results show small differences in the accumulated illumination, with a maximum difference of 0.0456%.

		Illumination (%)		
Latitude	Longitude	TU-Shader	TUB-Light	Difference (%)
-89.6868	197.2136	85.5134	85.5108	0.0026
-89.6873	197.3648	85.3214	85.2971	0.0243
-89.6866	197.3288	85.3211	85.2826	0.0385
-89.6874	197.2494	85.3109	85.2653	0.0456
-89.6862	197.1779	85.0304	85.0163	0.0141
-89.6879	197.4009	84.9049	84.8862	0.0187
-88.8092	123.6856	84.7757	84.7740	0.0017
-88.8088	123.7120	84.5291	84.5439	0.0148
-88.8083	123.6944	84.2531	84.2374	0.0157
-89.6885	197.4372	84.2131	84.2022	0.0109

TABLE 5.7: List of the 10 most illuminated spots for TUB-Shader and TUB-Light using floating points for an area of 50 x 50 km over a 20-year period at 2 m height.

As the input data used in both tools is the same, the source of these differences is believed to be caused during the calculation. TUB-Shader uses double precision while TUB-Light can work with float and double precision. The use of floats or doubles depends on the specifications of the platform where the simulation is run. Therefore, using TUB-Light in its float configuration, might cause small rounding errors when comparing to software running with double precision (see Table 5.7).

In order to quantify the differences between floating points and doubles, a number addition test was carried out. The test consists of a series of additions with different variable declarations stopping after the same number of iterations. The results of the addition example show a truncation error in the order of  $10^{-5}$ . Goldberg [1991] explains, that the precision loss leading to truncation errors is much easier to happen when using floats. This is due to the difference in the number of decimal digits existing in both declarations.

Additionally, a simulation for the same 50 x 50 km area using double precision declaration on another GPU was conducted. In Table 5.8 the differences between TUB-Shader and TUB-Light in its double configuration for the 10 most illuminated spots is presented. Although the errors still occur, they have been notably reduced. The remaining errors



might be produced by a difference in the calculation steps and processes compare to TU-Shader.

Illumination (%)				
Latitude	Longitude	TUB-Shader	TUB-Light	Difference (%)
-89.6868	197.2136	85.5134	85.5203	-0.0069
-89.6873	197.3648	85.3214	85.3303	-0.0089
-89.6866	197.3288	85.3211	85.3293	-0.0082
-89.6874	197.2494	85.3109	85.3203	-0.0094
-89.6862	197.1779	85.0304	85.0373	-0.0069
-89.6879	197.4009	84.9049	84.9130	-0.0081
-88.8092	123.6856	84.7757	84.7764	-0.0007
-88.8088	123.7120	84.5291	84.5294	-0.0003
-88.8083	123.6944	84.2531	84.2537	-0.0006
-89.6885	197.4372	84.2131	84.2225	-0.0094

TABLE 5.8: List of the 10 most illuminated spots for TUB-Shader and TUB-Light using double precision for an area of 50 x 50 km over a 20-year period at 2 m height.

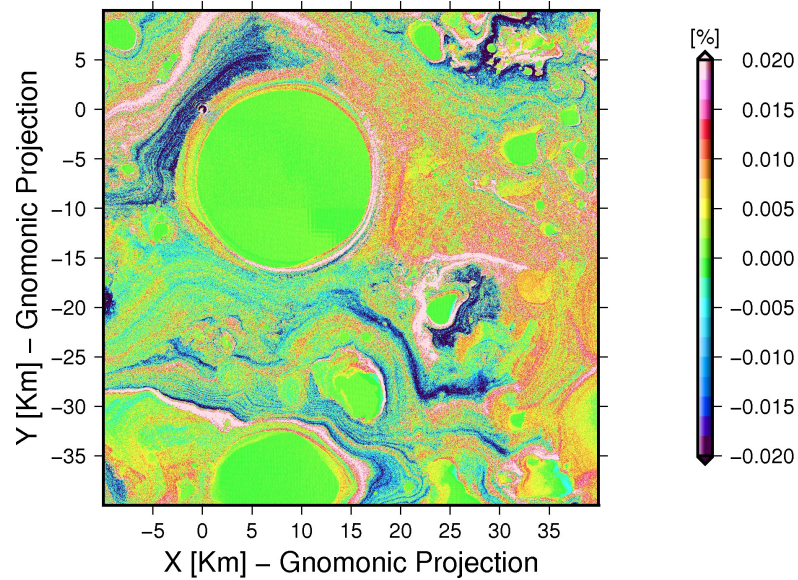
Figure 5.11 shows a graphic representation of the differences in Table 5.7 and Table 5.8.

The differences shown in Figure 5.11(a) are always larger than the differences in Figure 5.11(b). The areas with no differences are located mainly in the craters and PSA areas (see Figure 5.14). The 0% differences in these areas were expected since using both floating points and double precision configurations the accumulated illumination is 0%.

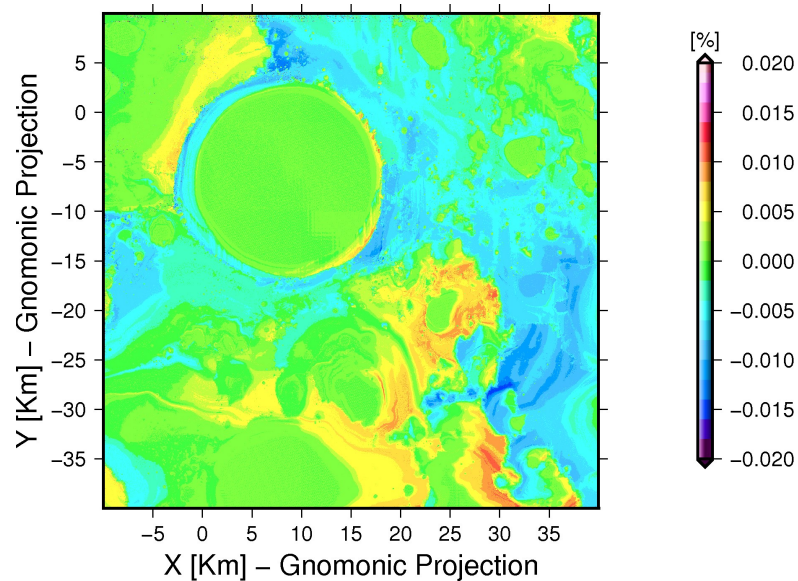
For Figure 5.12 the histograms for the values in Figure 5.11 are shown. In Figure 5.12(a) the histogram of the differences between TUB-Shader and TUB-Light in its floating point configuration is presented. Figure 5.12(b) shows the histogram of the differences between TUB-Shader and TUB-Light in its double precision configuration.

The histogram in Figure 5.12(a) reveals a balanced distribution of illumination differences around 0%. The maximum frequency of 20% corresponds to an illumination difference of the 0%. Figure 5.12(b) shows a similar distribution than Figure 5.12(a) however, the frequency is higher in values close to 0%. The maximum frequency of 25% occurs at a value of the 0%.

This test supports the theory that the major part of the differences comparing TUB-Shader and TUB-Light are caused by a floating point error. Those differences can be easily reduced by using TUB-Light in its double precision configuration.

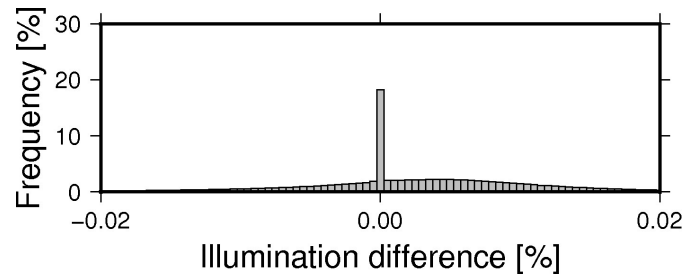


(a) Differences in accumulated illumination between TUB-Shader and TUB-Light in its floating point configuration

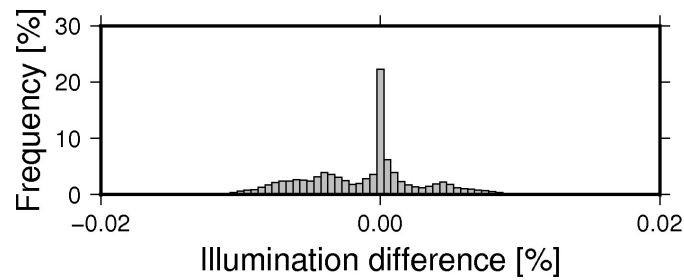


(b) Differences in accumulated illumination between TUB-Shader and TUB-Light in its double precision configuration

FIGURE 5.11: Comparison of illumination differences using floating points and double precision. Figure (a) shows the differences in illumination using floating points and Figure (b) the differences in illumination using double precision compared to TUB-Shader.



(a) Histogram of the differences in accumulated illumination between TUB-Shader and TUB-Light in its floating point configuration



(b) Histogram of the differences in accumulated illumination between TUB-Shader and TUB-Light in its double precision configuration

FIGURE 5.12: Histogram for the differences between TUB-Shader and TUB-Light using floating point and double precision configuration.

The floating point error distribution along time for a certain location ( $89.4394^{\circ}\text{S}$ ,  $222.8065^{\circ}\text{E}$ ) is shown in Figure 5.13. These errors are periodically distributed along the analyzed time period. We consider these errors to be noise created during the computation and therefore can be neglected.

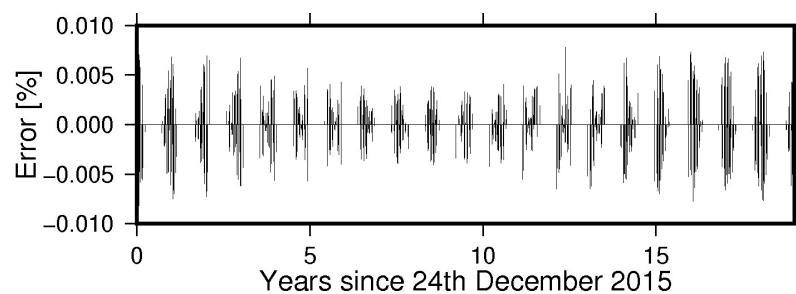


FIGURE 5.13: Remaining errors when comparing TUB-Shader and TUB-Light in double precision configuration.

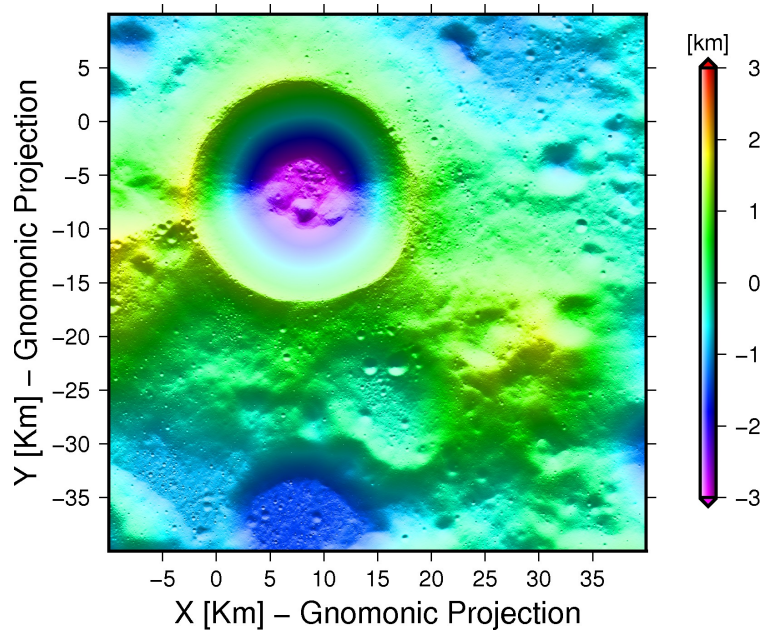


FIGURE 5.14: Topography for the 50 x 50 km analyzed area.

For Figure 5.15 the illumination differences using TUB-Light in its floating point and double precision configurations are shown. Here the differences are smaller than in Figure 5.11. The histogram of the illumination differences is shown in Figure 5.16. The histogram shows a normal distribution around 0%. The frequency in this histogram is higher in values close to 0% with a maximum value at this point of 38%.

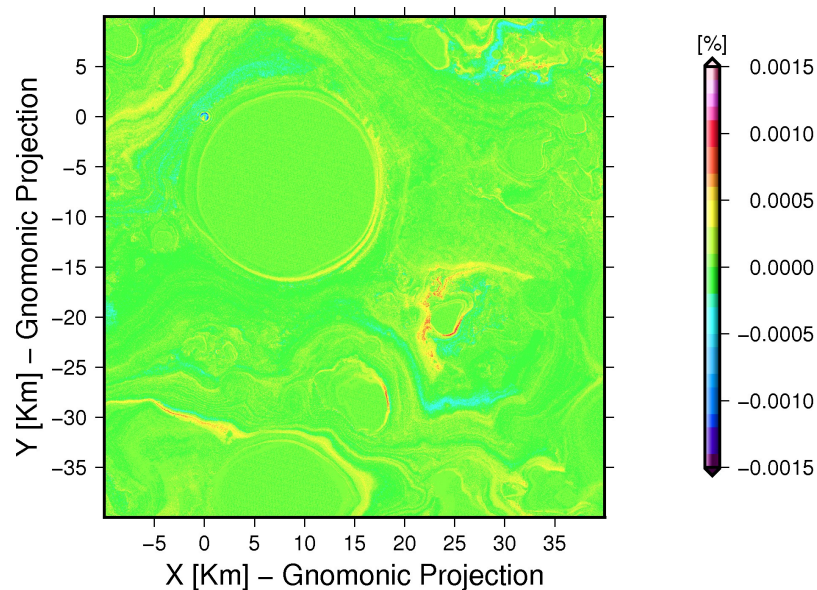


FIGURE 5.15: Comparison of TUB-Light with floating point and double precision configurations.

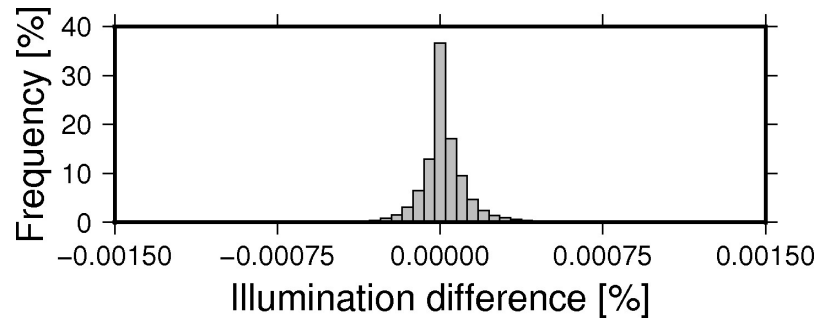


FIGURE 5.16: Histogram of TUB-Light with floating point and double precision configurations.

The maximum illumination difference over a 20-year period for the analyzed 50 x 50 km area is 3.3288 days using TUB-Light in its floating point configuration and 16.4688 hours using TUB-Light in its double precision configuration. To conclude, in order to reduce the differences between TUB-Shader and TUB-Light, the use of TUB-Light in its double precision configuration is recommended with respect to TUB-Light in its floating point configuration.

## Chapter 6

# Case Study I: Illumination studies for Luna-Resurs mission

### 6.1 Description of the Russian lunar mission: Luna-Resurs

Luna-Resurs (also called Luna-Resource and Luna-Glob 2) is an orbiter-rover mission planned to be launched around 2018. The main goal of the mission is to land near the lunar south pole to study frozen volatiles in craters. The nominal time for mission duration is one year, where the solar powered rover will travel to one of the PSAs (Slyuta et al. [2010]) analyzing possible water-ice volatiles in the sub-surface (Mitrofanov et al. [2011]). The lander (Figure 6.1) will carry a large amount of scientific payload. In Table 6.1 a list of the different instruments is presented.

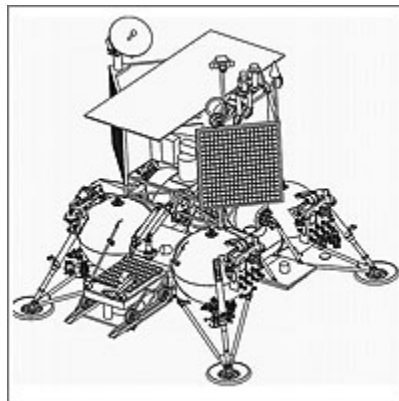


FIGURE 6.1: Concept view of the Luna-Resurs lander (RussianSpaceWeb. [Accessed: 24 october 2014]).

Instrument	Measurements/Operations	Organization
ADRON-LR	Active neutron and gamma-ray analysis of regolith	IKI
Analytic complex	Chromatographic and mass spectroscopy analysis of volatiles content and chemical composition	IKI
ARIES-L	Measurements of plasma of exosphere	IKI
LASMA-LR	Laser mass-spectrometer	IKI+ U. of Bern
LIS-TV-RPM	IR spectrometry of minerals and TV imaging	IKI
LINA	Measurements of plasma and neutrals	IKI+ ISP (Sweden)
PmL	Measurements of dust and micrometeorites	IKI
Radio-Beacon	Radio signal with very high stability	IKI
RAT	Radio measurements of thermal property of regolith	IKI
SEISMO-LR	Measurements of seismic activity	IFZ
TV-Spectrometer	UV and optical imaging of minerals with UV excitation	IKI
TERMO-L	Measurements of thermal properties of regolith	GEOKHI
STS-L	TV imaging of panoramas and area near Lander	IKI
Laser Retro Reflector	Moon libration and Moon ranging experiments	NPO SPP

TABLE 6.1: List of the payload on board the Luna-Resurs lander (RussianSpaceWeb. [Accessed: 24 october 2014]).

The candidate landing site for this mission is located in the western limb of Scott crater (see Figure 6.2) where PSA areas exist. An area of 40 x 40 km centered at 82.7°S, 33.5°E (see Figure 6.3) is studied in order to analyze its illumination conditions. We evaluated



a one-year period from February, 1 2018 to February, 1 2019 covering the nominal time of the mission.

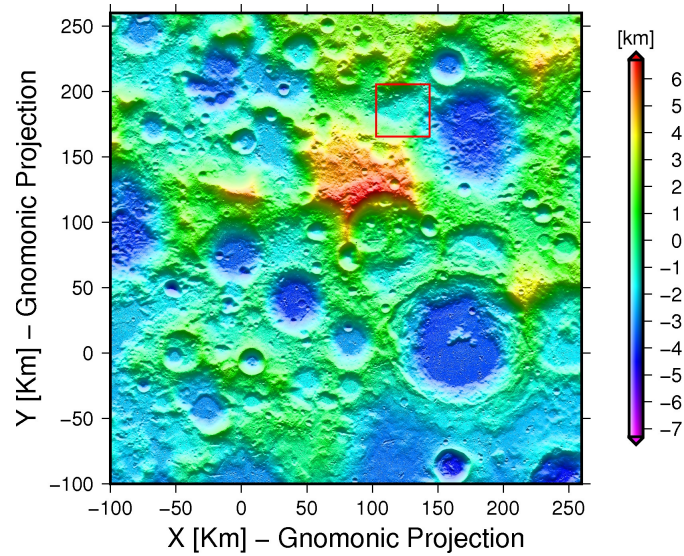


FIGURE 6.2: Topography of the lunar south pole with the candidate Luna-Resurs landing site marked by the red square.

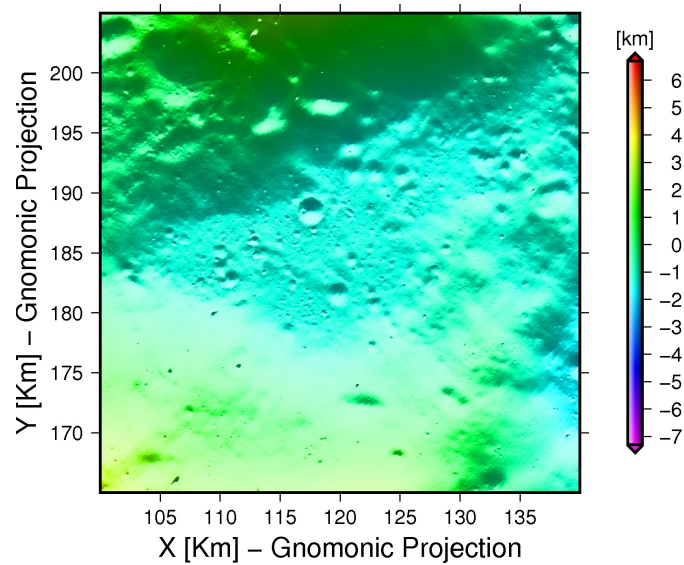


FIGURE 6.3: Topography of an area of 40 x 40 km centered at the Luna-Resurs landing site.



## 6.2 Data

The data used in this investigation is a set of horizons containing elevations of the highest terrain features in an area of 40 x 40 km with a resolution of 20 m/pixel at 2 m height. The DTM used to calculate the horizons is an adjusted LOLA DTM with a resolution of 20 m/pixel and covering a region of 300 x 300 km Gläser et al. [2014a].

The considered area lies 200 km away from the south pole where LOLA tracks density is lower than near the pole. This results in a product with interpolated data and where artifacts might appear. Interpolated areas can be noticed around  $x = 135$  km and  $y = 180$  km in Figure 6.3. Furthermore, artifacts in the horizons derived from the DTM can be seen around azimuth  $320^\circ$  in Figure 6.5. As this is an inherent problem, it might induce small errors in the results.

## 6.3 Results

In Figure 6.4 the accumulated illumination for an area of 40 x 40 km over a 20-year period at 2 m height is presented. The candidate landing site area receives an accumulated illumination of 24.4428% on average, with a maximum of 53.8634%.

The reduced illumination can be explained by analyzing the trajectory of the Sun vs. the horizon (Figure 6.5). Figure 6.5 reveals that the Sun is below the horizon approximately the 80% of the time. The trajectory of the Sun results in a very repetitive Illumination pattern (Figure 6.6), which quantifies the amount of the visible solar-disk over the analyzed period. The topography of the analyzed area (Figure 6.2) reveals that the candidate landing site is located at a height of -1 km and appears to be surrounded by terrain features higher than 6 km. This circumstances results in a landing site with a reduced illumination.

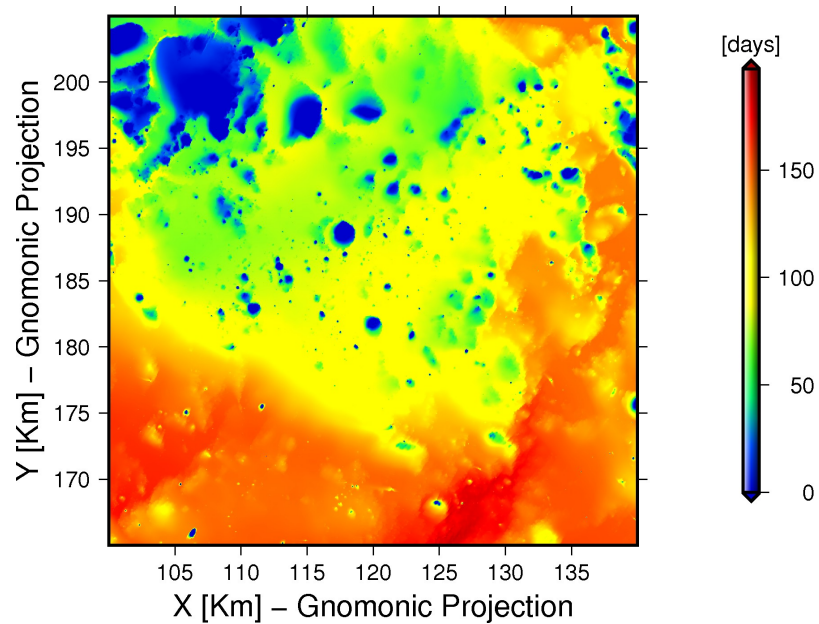


FIGURE 6.4: Accumulated Illumination for an area of 40 x 40 km over a 20-year period at 2 m height..

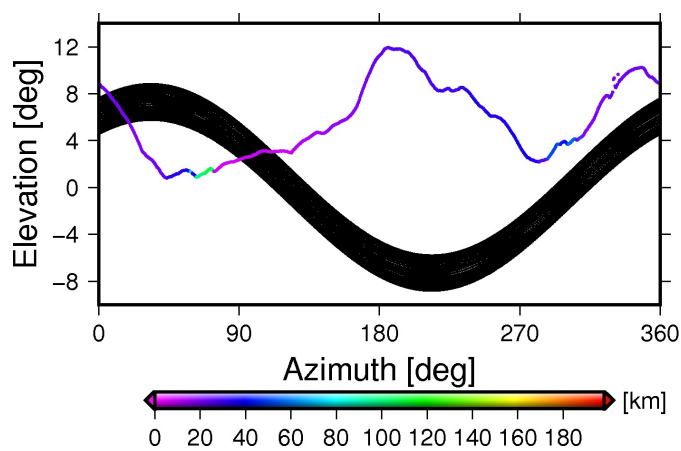


FIGURE 6.5: Trajectory of the Sun and shape of the horizon as seen from  $82.6999^{\circ}\text{S}$ ,  $33.502^{\circ}\text{E}$ . Black corresponds to the Sun. The horizon is represented color-coded the distance to the highest elevated feature.

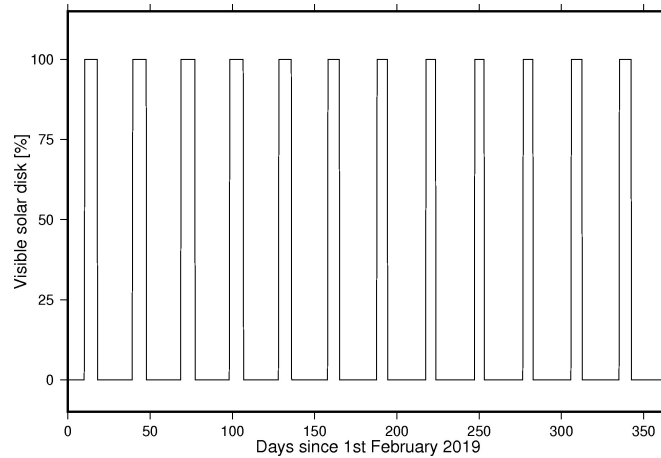


FIGURE 6.6: Illumination pattern located at  $82.6999^{\circ}\text{S}$ ,  $33.502^{\circ}\text{E}$  over a one-year period at 2 m height.

The results obtained in this investigation cannot be cross-validated, so the conclusions are built considering the outcome of this study. Based on the analysis of the topography of the candidate landing site, the shape of Sun's trajectory and the derived accumulated illumination, we conclude that the results seem to be consistent.

Unlike the lander described in Carpenter et al. [2012], the lander of the Luna-Resurs mission is not using solar arrays. Therefore, a scenario with a large accumulated illumination is not a priority of the mission. As the main goal of the mission is to find water-ice, a reduced illumination is a key characteristic to consider this area as a perfect candidate landing site. This, combined with the fact that PSAs exist in the vicinity of the landing site (Slyuta et al. [2010]), leads us to consider this area a satisfactory landing site for the achievement of the objectives of the mission.

## Chapter 7

# Case Study II: Illumination conditions at Shackleton Vicinity candidate Landing site

### 7.1 Description of Shackleton Vicinity candidate Landing site

The Shackleton Vicinity (SV1) candidate landing site is located in the surroundings of the Shackleton crater. This area lies within the South Pole-Aitken basin. The center of SV1 is located at  $85.5^{\circ}\text{S}$ ,  $37.4^{\circ}\text{E}$ . In Figure 7.1 the SV1 candidate landing site area is shown. The candidate landing site lies approximately 15 km north-east Shackleton crater. Figure 7.1 reveals that the candidate landing site lies in an elevated area.

SV1 is one of the most illuminated regions at the lunar south pole. The existence of PSAs and the possibility of finding water-ice, appoints this area to be of great interest. SV1 was selected as a candidate landing site in the study carried out by De Rosa et al. [2012].

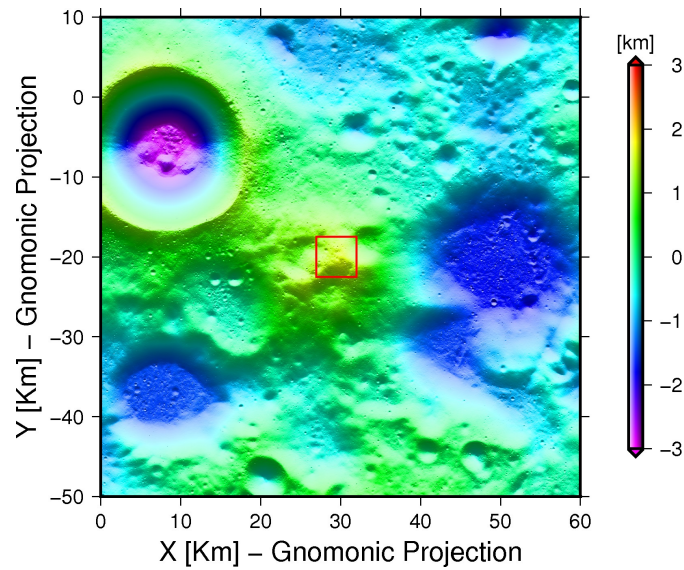


FIGURE 7.1: Location of the Shackleton Vicinity. The landing site area is marked within a red square. Note that the south pole is located in the upper-left part of the figure.

## 7.2 Data

The data used in this investigation is a set of horizons containing elevations of the highest terrain features in an area of 50 x 50 km with a resolution of 20 m/pixel at 2 m height. The DTM used to calculate the horizons is an adjusted LOLA DTM with a resolution of 20 m/pixel and covering a region of 300 x 300 km Gläser et al. [2014a].

## 7.3 Results

Figure 7.2 shows the accumulated illumination over a 20-years period at 2 m height. An area located around  $x = 30$  km and  $y = -20$  km receives the highest illumination (Figure 7.3). A list of the 10 most illuminated spots is shown in Table 7.1.

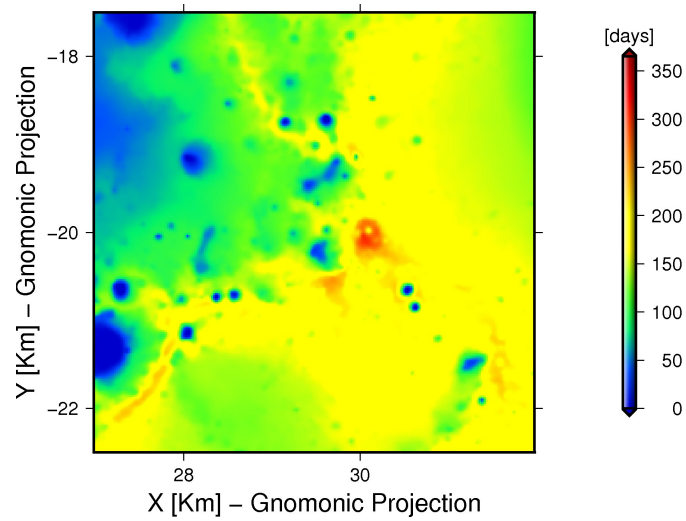


FIGURE 7.2: Accumulated illumination for SV1 over a 20-year period at 2 m height.

Latitude	Longitude	Accumulated illumination %
-88.8092	123.6856	84.7740
-88.8088	123.7120	84.5439
-88.8083	123.6944	84.2374
-88.8093	123.7296	83.8299
-88.8084	123.7384	83.7785
-88.8097	123.7032	83.6287
-88.8086	123.6680	83.6067
-88.8079	123.7208	83.4729
-88.8090	123.7560	83.2922
-88.8075	123.7471	83.0106

TABLE 7.1: List of the 10 most illuminated spots for SV1 over a 20-year period at 2 m height.

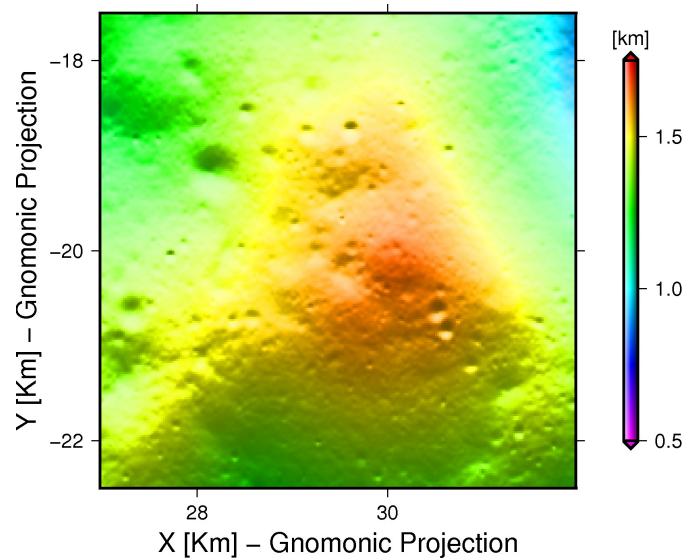


FIGURE 7.3: Topography of SV1 candidate landing site.

According to Bussey et al. [2010], a spot with an accumulated illumination of 86% is located at  $88.74^{\circ}\text{S}$ ,  $124.5^{\circ}\text{E}$  which corresponds to SV1. The study was conducted using DTMs derived from Kaguya data with a resolution of 470 m/pixel, and covering the year 2020 with a time step of 12 hours. The method used to calculate illumination was the ray-tracing method. The different configuration, methods and data might cause variations compared to the results in this investigation.

In De Rosa et al. [2012], a spot located at  $88.8014^{\circ}\text{S}$ ,  $123.69^{\circ}\text{E}$  receives an accumulated illumination of 83.16% (using the ESA Coverage tool) and 80.5085% (using the ASP Lunar-Shader tool). Both tools used a 240 m/pixel resolution DTM. The analyzed period is from March 31, 2010 at 12:00 hours to September 24, 2010 at 00:00 hours.

In this investigation we obtained illumination of 65.9265% for a spot located at  $88.8057^{\circ}\text{S}$ ,  $123.5365^{\circ}\text{E}$  using a 20 m/pixel resolution DTM at 0 m height. Using a DTM with 20 m/pixel resolution at 2 m height, we obtained illumination of 84.7740% for a spot located at  $88.8092^{\circ}\text{S}$ ,  $123.6856^{\circ}\text{E}$ . A summary of the results obtained from the comparison is shown in Table 7.2.

Study	Res.[m/pix]	Illu.[%]	Location latitude, longitude	h [m]	Period [y]
Noda et al. 2008	474	86	-88.8, 124.1	0	1pc
Bussey et al. 2010	474	86	-88.74, 124.5	0	1pc
De Rosa et al. 2012	240	83.16	-88.8014, 123.69	0	0.5
De Rosa et al. 2012	240	80.5085	-88.8014, 123.69	0	0.5
This study	20	65.9265	-88.8057, 123.5356	0	20
This study	20	84.7740	-88.8092, 123.6856	2	20

TABLE 7.2: List of the most illuminated points in SV1 from previous studies with the one found in this study. The analyzed period in De Rosa et al. [2012] was 6 months, from 31 March, 2010 to 24 September, 2010. The "1pc" period corresponds to a lunar precessional cycle (18.6 years).

Despite the low resolution DTMs used in Noda et al. [2008] and Bussey et al. [2010], the obtained results are very close to De Rosa et al. [2012]. If we compare the results to this study at 0 m height, a difference of 20% is achieved. These differences can be caused by the fact that, compared to Noda et al. [2008] and Bussey et al. [2010], the adjusted DTMs in this study have a 20 times more resolution. The obtained differences are in the same level as presented in Gläser et al. [2014b]. When the illumination is calculated at 2 m height, the results are similar to Noda et al. [2008] and Bussey et al. [2010].

In Figure 7.4 the trajectory of the Sun and the horizon as seen from  $88.8092^{\circ}\text{S}$ ,  $123.6856^{\circ}\text{E}$  is presented. Figure 7.4 reveals that the Sun lies above the horizon for the majority of the time during the analyzed period, providing long periods of illumination. The Sun pattern in Figure 7.5, reveals the longest period in constant darkness to be of 5.75 days and the longest period in constant illumination of 146.25 days.

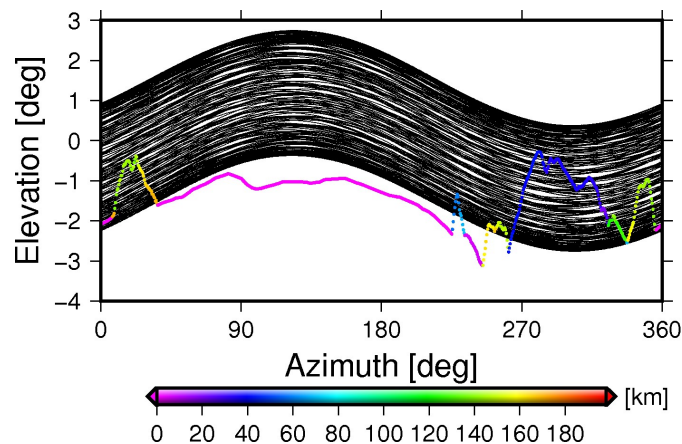


FIGURE 7.4: Trajectory of the Sun and shape of the horizon for SV1. Black corresponds to the Sun. The horizon is color-coded the distance of the highest elevated feature.



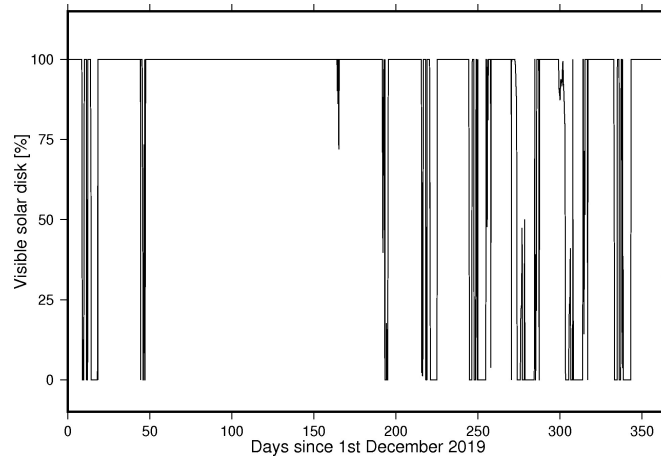


FIGURE 7.5: Illumination pattern as seen from  $88.8092^{\circ}\text{S}$ ,  $123.6856^{\circ}\text{E}$  over a one-year period at 2 m height.

We conclude that, the SV1 area can be considered a good candidate landing site. The maximum illumination is 84.7740% and 1% of the spots receive more than 300 days of accumulated illumination. In Figure 7.3 a large smooth area can be seen. The largest height difference at the analyzed area is in the order of 1 m, therefore a lander can safely land in this area. The lander described in Carpenter et al. [2012], using hazard avoidance navigation techniques to land with more precision, might consider this area as a safe landing site.

Due to the topography of the candidate landing site PSAs exist, with the possibility of harboring water-ice. PSAs lie within a distance of 2 km. A rover traveling to these areas will receive more than 200 days of illumination. If the Sun patterns of the rover tracks are analyzed, a detailed schedule to visit these areas can be calculated. Due to time restriction of this investigation, the Sun pattern analysis of the rover tracks is left for future investigations.

## Chapter 8

# Conclusions and Outlook

In this work, a parallel computing tool for the evaluation of illumination conditions at the lunar south pole was developed and tested. The tool is capable of achieving results comparable to previous studies reducing the computation time by a factor of 60. Different tests and cross-validations were developed to proof the reliability of the tool. Those results were compared to previous and recent studies. All results were found to be in accordance.

The analysis of illumination for the Luna-Resurs candidate landing site reveals that, for the goals of the mission, this area can be considered as a satisfactory site. The existence of PSAs (Slyuta et al. [2010]) and the possibility that they might harbor water-ice, offers a great opportunity for its investigation.

For the SV1 candidate landing site, the results reveal that it can be considered as a satisfactory landing site due to its high accumulated illumination and its smooth topography. Additionally, the highly illuminated landing site can provide enough solar power for the correct functioning of a lander. The possibility of finding water-ice nearby, is also considered an asset for the selection of SV1 as a satisfactory candidate landing site.

Several possible extensions to follow-on with this work are considered. A better implementation of the plotting process, specially when selecting the coordinates of the observer to plot the Sun pattern. The tool sorts the Sun azimuths in order to reduce time the time spend in the open/close file sequence. The sorting algorithm in TUB-Light is the so-called "Bubble sort". The "Bubble Sort" algorithm is highly time consuming

and depends on how long is the period which needs to be analyzed, therefore a new parallel implementation of the algorithm should be included.

TUB-Light computes illumination conditions but is not yet capable to calculate horizons. An extension for the horizon calculation should be created in order to reduce the computation time needed in sequential programming techniques. This approach has been tested but, due to time and expertise reasons, was decided to left unfinished.

Lunar exploration is of interest for different space agencies. New missions are going to be launched in the following years starting with the Russian Luna-Glob in 2016 (Exploring Space. [Accessed: 9 october 2014]) and the Indian Chandrayaan-2 in 2016-2017 (Hindustan times. [Accessed: 9 october 2014]). Some of these missions are landers, some orbiters and some sample return missions. In the last few years the private sector started to invest in lunar exploration missions. This triggered the creation of some private companies like Astrobotic Technology or Shackleton Energy Company. The goal for Astrobotic Technology is to be the first private company to successfully launch a mission with a lander and a rover to the moon. The aim of Shackleton Energy Company is to prepare the equipment and technologies necessary for mining the Moon.

Also the X prize foundation together with Google has launched a space competition to successfully launch a spacecraft, fly to the Moon, land safely on the surface and travel the lunar surface with a rover while sending images to Earth.

This leads to the final conclusion that it is necessary to keep investigating the Moon in general and, in the scope of possible future landers and lunar bases, illumination conditions in particular.

# Appendix A

## TUB-Light: User manual

### A.1 Requirements

The system requires several files before launching the tool. A set of configuration files are required in order to start the tool, which will be described in Appendix A.2. These configuration files contain a set of initialization fields necessary for the correct functioning of the tool.

TUB-Light uses third party APIs, libraries and software, which need to be correctly installed and configured on the computer (Table A.1). Not all GPUs and CPUs will support OpenCL. It mainly depends on the age and the model of the hardware. For this reason a small script is included in the package in order to check if the tool can be run on the selected computer.

Name	Purpose	Version
OpenCL	Parallel computing	1.0
CSpice	Planetary ancillary data	N0065

TABLE A.1: List of APIs, libraries and software needed.

### A.2 Configuration Files

A configuration file is composed by a set of initial parameters necessary to run the tool. The initial parameters are the start and stop time of the analysis, number of pixels of

the RoI, time stamp, source of the data, path to horizon files, path to a file with the coordinates of the RoI and, the spot from where the Sun pattern will be analyzed.

### A.2.1 Structure of Configuration Files

The configuration file keeps a certain structure. Following this structure ensures the correct operation of the tool. Figure A.1 shows the structure of a configuration file.

```

1 -----
2 THIS IS THE CONFIG FILE FOR THE ILLUMINATION TOOL GPGPU
3 NO CHANGE FORMAT
4 -----
5
6 2018-OCT-22-00:00:00                               ///START TIME
7 2019-OCT-22-00:00:00                               ///STOP TIME
8 10000                                               ///NUMBER OF PIXELS
9 0                                                    ///TIME STAMP YES(1) OR NO(0)
10 0                                                  ///PHIL DATA(1) OR RAMIRO DATA(0)
11 /users/figu_ra/workspace/interp_test/CR1_2km_2km_2m/  ///PATH TO HORIZON FILES
12 /users/figu_ra/Desktop/plot_stuff/illumination_plots/accumulated/CR1_2kmx2km/records.txt  ///PATH TO FILE OF RoI
13 5050                                               ///PATTERN
14 0                                                    ///CONFIGURATION: (0)FLOAT (1)DOUBLE |

```

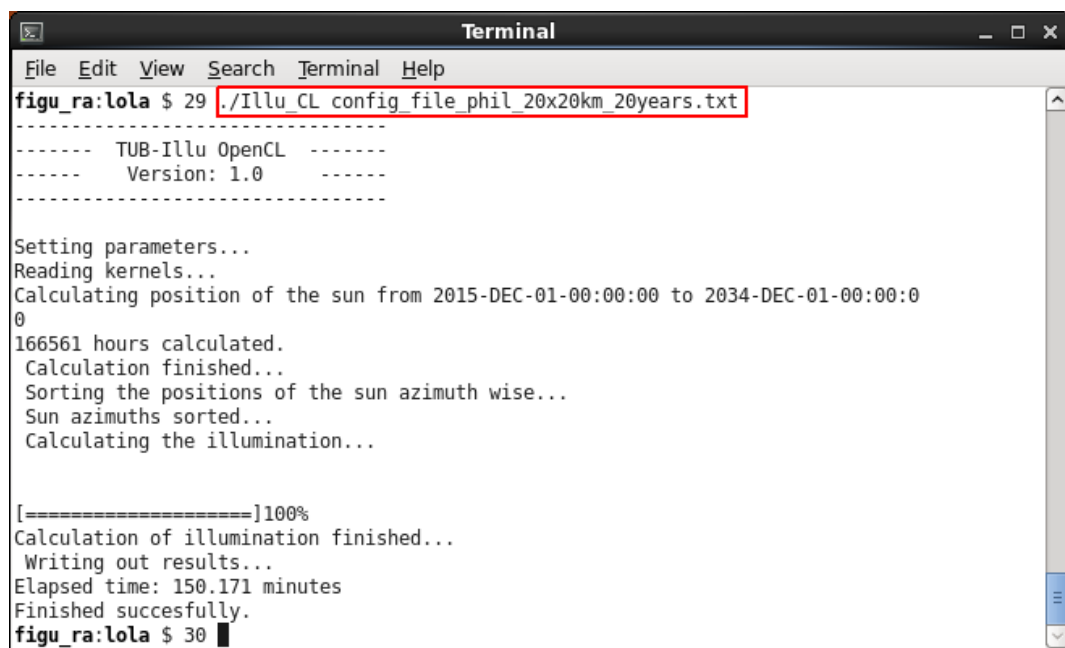
FIGURE A.1: Example of the structure of a configuration file structure.

- **START TIME:** This field contains the starting UTC time of the simulation.
- **STOP TIME:** This field contains the stopping UTC time of the simulation.
- **NUMBER OF PIXELS:** This field contains the number of pixels in the RoI. This can be calculated by the resolution of the input data.
- **TIME STAMP:** This field selects whether a time stamp will be created or not. This parameter, if selected (set to 1), will create files containing illumination conditions for each time step. It has to be considered that it will create as many files as time steps in the analysis. As an example, for a one-year period with a time step of 1 hour, the tool will create 8760 files corresponding to 8760 hours in a year. This process will affect the performance of the tool since it involves a large number of open/close file operations. Moreover, depending on the characteristics of the computer and the analyzed period (e.g. 20 years analysis), the creation of a large number of files might freeze the computer.
- **DATA FORMAT:** This field selects the data format. The tool uses two different data format, depending on the tool where the horizon files were created.
- **PATH TO HORIZON DIRECTORY:** This field contains the absolute path to the directory where the horizon files are located.

- **PATH TO FILE OF RoI:** This field contains the absolute path to the file containing the coordinates of the RoI.
- **PATTERN:** This field contains the entry in the RoI file to analyze the Sun pattern.
- **CONFIGURATION:** This field selects the configuration of the tool. There are two options, floating points configuration (0) or double precision configuration (1). In case the double precision configuration is selected, the information contained in Appendix B.1.4 should be considered.

### A.3 Execution

The tool is an executable file. The configuration file should be sent as an argument to the tool for its correct functioning (Figure A.2).



```
Terminal
File Edit View Search Terminal Help
figu_ra:lola $ 29 ./Illu_CL config_file_phil_20x20km_20years.txt
----- TUB-Illu OpenCL -----
----- Version: 1.0 -----
-----

Setting parameters...
Reading kernels...
Calculating position of the sun from 2015-DEC-01-00:00:00 to 2034-DEC-01-00:00:0
0
166561 hours calculated.
Calculation finished...
Sorting the positions of the sun azimuth wise...
Sun azimuths sorted...
Calculating the illumination...

[=====]100%
Calculation of illumination finished...
Writing out results...
Elapsed time: 150.171 minutes
Finished succesfully.
figu_ra:lola $ 30 █
```

FIGURE A.2: Example of how the tool is executed.

# Appendix B

## Developers Manual

### B.1 APIs, libraries and toolkits used

#### B.1.1 OpenCL

Open Computing Language<sup>1</sup> (Gaster et al. [2008] and Munshi et al. [2011]) is a framework used to write and execute parallel programming code across different platforms. Depending on the hardware's vendor specific drivers will need to be installed for the correct functioning of OpenCL.

#### B.1.2 Spice Toolkit

Spice Toolkit<sup>2</sup> (NAIF. [Accessed: 10 october 2014]) is an information system, developed at the NASA Navigation and Ancillary Information Facility (NAIF), to assist interpreting scientific observations from space-borne instrument and modeling planetary missions. It can be run under different languages. TUB-Light uses its C extension called C-spice.

#### B.1.3 GMT

GMT<sup>3</sup> is an open source collection of command-line tools for manipulating and plotting geographic data sets.

---

<sup>1</sup>Available at: <http://www.khronos.org/opencv/>

<sup>2</sup>Available at: [http://naif.jpl.nasa.gov/naif/toolkit\\_C.html](http://naif.jpl.nasa.gov/naif/toolkit_C.html).

<sup>3</sup>Available at: <http://gmt.soest.hawaii.edu/projects/gmt/wiki/Download>

### B.1.4 TUB-Light in double precision configuration

TUB-Light can be configured to be run using double precision. Before selecting this configuration a script called "check\_compatibility.cpp" has to be launched to inquire whether the platform will support double precision or not. Figure B.1 shows the output of the script. In the field inside the red box a 1 will appear if the platform supports double precision.

```

Terminal
File Edit View Search Terminal Help
Number of platforms: 1
CL_PLATFORM_PROFILE: FULL_PROFILE
CL_PLATFORM_VERSION: OpenCL 1.1 CUDA 6.0.1
CL_PLATFORM_VENDOR: NVIDIA Corporation
CL_PLATFORM_EXTENSIONS: cl_khr_byte_addressable_store cl_khr_icd cl_khr_gl_sharing cl_nv_compiler_options cl_nv_device_attribute_query cl_nv_pragma_unroll
Number of devices: 1
CL_DEVICE_TYPE: CL_DEVICE_TYPE_GPU
CL_DEVICE_VENDOR_ID: 4318
CL_DEVICE_MAX_COMPUTE_UNITS: 2
CL_DEVICE_MAX_WORK_ITEM_DIMENSIONS: 3
CL_DEVICE_MAX_WORK_ITEM_SIZES: 1024 1024 64
CL_DEVICE_MAX_WORK_GROUP_SIZE: 1024
CL_DEVICE_PREFERRED_VECTOR_WIDTH_CHAR: 1
CL_DEVICE_PREFERRED_VECTOR_WIDTH_SHORT: 1
CL_DEVICE_PREFERRED_VECTOR_WIDTH_INT: 1
CL_DEVICE_PREFERRED_VECTOR_WIDTH_LONG: 1
CL_DEVICE_PREFERRED_VECTOR_WIDTH_FLOAT: 1
CL_DEVICE_PREFERRED_VECTOR_WIDTH_DOUBLE: 1
CL_DEVICE_PREFERRED_VECTOR_WIDTH_HALF: 0
CL_DEVICE_NATIVE_VECTOR_WIDTH_CHAR: 1
CL_DEVICE_NATIVE_VECTOR_WIDTH_SHORT: 1
CL_DEVICE_NATIVE_VECTOR_WIDTH_INT: 1
CL_DEVICE_NATIVE_VECTOR_WIDTH_LONG: 1
CL_DEVICE_NATIVE_VECTOR_WIDTH_FLOAT: 1
CL_DEVICE_NATIVE_VECTOR_WIDTH_DOUBLE: 1
CL_DEVICE_NATIVE_VECTOR_WIDTH_HALF: 0
CL_DEVICE_MAX_CLOCK_FREQUENCY: 1200
CL_DEVICE_ADDRESS_BITS: 32
CL_DEVICE_MAX_MEM_ALLOC_SIZE: 268222464
CL_DEVICE_IMAGE_SUPPORT: 1
CL_DEVICE_MAX_READ_IMAGE_ARGS: 128
CL_DEVICE_MAX_WRITE_IMAGE_ARGS: 8
CL_DEVICE_IMAGE2D_MAX_WIDTH: 32768
CL_DEVICE_IMAGE2D_MAX_HEIGHT: 32768
CL_DEVICE_IMAGE3D_MAX_WIDTH: 2048
CL_DEVICE_IMAGE3D_MAX_HEIGHT: 2048
CL_DEVICE_IMAGE3D_MAX_DEPTH: 2048
CL_DEVICE_MAX_SAMPLERS: 16
CL_DEVICE_MAX_PARAMETER_SIZE: 4352
CL_DEVICE_MEM_BASE_ADDR_ALIGN: 4096
CL_DEVICE_MIN_DATA_TYPE_ALIGN_SIZE: 128
CL_DEVICE_SINGLE_FP_CONFIG: CL_FP_DENORM | CL_FP_INF_NAN | CL_FP_ROUND_TO_NEAREST | CL_FP_ROUND_TO_ZERO | CL_FP_ROUND_TO_INF | CL_FP_FMA
CL_DEVICE_GLOBAL_MEM_CACHE_TYPE: CL_READ_WRITE_CACHE
CL_DEVICE_GLOBAL_MEM_CACHE_SIZE: 128
CL_DEVICE_GLOBAL_MEM_SIZE: 32768
CL_DEVICE_MAX_CONSTANT_BUFFER_SIZE: 1072889856
CL_DEVICE_MAX_CONSTANT_ARGS: 9
CL_DEVICE_LOCAL_MEM_TYPE: CL_LOCAL_MEM_TYPE
CL_DEVICE_LOCAL_MEM_SIZE: 49152
CL_DEVICE_ERROR_CORRECTION_SUPPORT: 0
CL_DEVICE_HOST_UNIFIED_MEMORY: 0
CL_DEVICE_PROFILING_TIMER_RESOLUTION: 1000
CL_DEVICE_ENDIAN_LITTLE: 1
CL_DEVICE_AVAILABLE: 1
CL_DEVICE_COMPILER_AVAILABLE: 1
CL_DEVICE_EXECUTION_CAPABILITIES: CL_EXEC_KERNEL
CL_DEVICE_QUEUE_PROPERTIES: CL_QUEUE_OUT_OF_ORDER_EXEC_MODE_ENABLE | CL_QUEUE_PROFILING_ENABLE
CL_DEVICE_PLATFORM: 0x1e89250
CL_DEVICE_NAME: Quadro 600
CL_DEVICE_VENDOR: NVIDIA Corporation
CL_DRIVER_VERSION: 331.49
CL_DEVICE_PROFILE: FULL_PROFILE

```

FIGURE B.1: Output of the script to check if the selected GPU supports double precision.



# Bibliography

- H. Araki, S. Tazawa, H. Noda, Y. Ishihara, S. Sasaki, S. Goossens, N. Kawano, I. Kamiya, H. Otake, J. Oberst, and C. K. Shum. A new lunar topographic map of the moon by the laser altimeter (LALT) on board KAGUYA. *Science*, Vol. 323: 897–900, 2009.
- H. Araki, S. Sasaki, Y. Ishihara, H. Noda, H. Hanada, K. Matsumoto, F. Kikuchi, S. Goossens, N. Namiki, and T. Iwata. The first global topography and gravimetry of the moon by KAGUYA. *EPSC abstracts*, 2010.
- Argonne National Laboratory. Ibm blue gene/p. <http://www.flickr.com/photos/35734278@N05/3323018571/>, Accessed: 05 november 2014. URL <http://www.flickr.com/photos/35734278@N05/3323018571/>.
- D.B.J. Bussey, J.A. McGovern, P.D. Spudis, C.D. Neish, H. Noda, Y. Ishihara, and S.A. Sørensen. Illumination conditions of the south pole of the moon derived using kaguya topography. *Icarus*, 2010.
- J.D. Carpenter, R. Fisackerly, D De Rosa, and B Houdou. Scientific preparations for lunar exploration with the european lunar lander. *Planetary and Space Science*, Vol. 74:208–223, 2012.
- D. R. Williams, NASA (National Space Science Data Center). Moon fact sheet. <http://nssdc.gsfc.nasa.gov/planetary/factsheet/moonfact.html>, Accessed: 05 november 2014. URL <http://nssdc.gsfc.nasa.gov/planetary/factsheet/moonfact.html>.
- D. De Rosa, B. Bussey, J. T. Cahill, T. Lutz, I. A. Crawford, T. Hackwill, S. van Gasselt, G. Neukum, L. Witte, A. McGovern, P. M. Grindrod, and J. D. Carpenter.

- Characterisation of potential landing sites for the european space agency's lunar lander project. *Planetary and Space Science*, Vol. 74:224–246, 2012.
- Exploring Space. First russian moon mission delayed. <http://spaceexp.tumblr.com/post/64287200145/first-russian-moon-mission-delayed>, Accessed: 9 october 2014. URL <http://spaceexp.tumblr.com/post/64287200145/first-russian-moon-mission-delayed>.
- B. Gaster, L. Howes, D. R. Kaeli, P. Mistry, and D. Schaa. *Heterogeneous Computing with OpenCL*. Morgan Kaufmann, 2008.
- J. Ghorpade, J. Parande, M. Kulkarni, and A. Bawaskar. Gpgpu processing in cuda architecture. *Advanced Computing: An International Journal*, Vol. 3, 2012.
- P. Gläser, F. Scholten, I. Haase, J. Oberst, D. de Rosa, M. S. Robinson, G. A. Neumann, E. Mazarico, D. E. Smith, and M. T. Zuber. Improvement of local LOLA DTMs using LROC NAC DTMs - example for an ESA lunar lander candidate landing site. In *44th Lunar and Planetary Science Conference.*, 2014a.
- P. Gläser, F. Scholten, D. De Rosa, R. Marco Figuera, J. Oberst, E. Mazarico, G. A. Neumann, and M. S. Robinson. Illumination conditions at the lunar south pole using high resolution digital terrain models from lola. *Icarus*, 2014b.
- D. Goldberg. What every computer scientist should know about floating-point arithmetic. *ACM Computing Surveys*, Vol. 23, 1991.
- S. Gupta and R. Babu. Performance analysis of gpu compared to dingle-core and multi-core cpu for natural language applications. *Advanced Computer Science and Applications*, Vol. 2, 2011.
- Hindustan times. India to launch chandrayaan- ii by 2016-17. <http://www.hindustantimes.com/technology/science/india-to-launch-chandrayaan-ii-by-2016-17/article1-1171493.aspx>, Accessed: 9 october 2014. URL <http://www.hindustantimes.com/technology/science/india-to-launch-chandrayaan-ii-by-2016-17/article1-1171493.aspx>.

- K. C. Laurini, B. Hufenbach, J. Kawaguchi, J. Piedboeuf, B. Schade, J. Curtis, and H. Kim. An international strategy for human exploration of the moon: The international space exploration coordination group (isecg) reference architecture for human lunar exploration. In *61st International Astronautical Congress.*, 2010.
- R. Marco Figuera, P. Gläser, J. Oberst, and D. de Rosa. Calculation of illumination conditions at the lunar south pole - parallel programming approach. *EPSC abstracts*, Vol. 9, 2014.
- J. L. Margot, D. B. Campbell, R. F. Jurgens, and M. A. Slade. Topography of the lunar poles from radar interferometry: A survey of cold trap locations. *Science*, Vol. 284:1658, 1999.
- E. Mazarico, G. A. Neumann, D. E. Smith, M. T. Zuber, and M. H. Torrence. Illumination conditions of the lunar polar regions using LOLA topography. *Icarus*, Vol. 211: 1066–1081, 2011. Issue 2.
- I.G. Mitrofanov, L.M. Zelenyi, V.I. Tret'yakov, and V.P. Dolgoplov. Science program of lunar landers of "luna-glob" and "luna-resource" missions. In *42nd Lunar and Planetary Science Conference.*, 2011.
- A. Munshi, B. Gaster, T. G. Mattson, J. Fung, and D. Ginsburg. *OpenCL Programming Guide*. Addison-Wesley Professional, 2011.
- NAIF. The spice toolkit. <http://naif.jpl.nasa.gov/naif/toolkit.html>, Accessed: 10 october 2014. URL <http://naif.jpl.nasa.gov/naif/toolkit.html>.
- H. Noda, H. Araki, S. Goossens, Y. Ishihara, K. Matsumoto, S. Tazawa, N. Kawano, and S. Sasaki. Illumination conditions at the lunar polar regions by KAGUYA(SELENE) laser altimeter. *Geophysical Research Letters*, 35, 2008.
- NVIDIA. Cuda gpus. <https://developer.nvidia.com/cuda-gpus>, Accessed: 17 September 2013. URL <https://developer.nvidia.com/cuda-gpus>.
- RussianSpaceWeb. The design of the luna-resurs lander. [http://www.russianspaceweb.com/luna\\_resurs.html](http://www.russianspaceweb.com/luna_resurs.html), Accessed: 24 october 2014. URL [http://www.russianspaceweb.com/luna\\_resurs.html](http://www.russianspaceweb.com/luna_resurs.html).

- Scientific Volume Imaging. Micro-rotation workbench.  
<http://www.svi.nl/MicroRotationWorkbench>, Accessed: 24 october 2014. URL  
<http://www.svi.nl/MicroRotationWorkbench>.
- S. Seo, J. Lee, G. Jo, and J. Lee. Automatic opencl work-group size selection for multi-core cpus. In *22nd international conference on Parallel architectures and compilation techniques.*, 2013.
- E. N. Slyuta, A. M. Abdrakhimov, A. T. Basilevsky, E. N. Lazarev, V. P. Dolgoplov, and A. I. Sheikhet. Landing sites for the russian luna-resurs mission to the moon. In *41st Lunar and Planetary Science Conference.*, 2010.
- D. E. Smith, M. T. Zuber, G. A. Neumann, F. G. Lemoine, E. Mazarico, M.H. Torrence, J. F. McGarry, D. D. Rowlands, J. W. Head, T. H. Duxbury, O. Aharonson, P. G. Lucey, M-S. Robinson, O. S. Barnouin, J-F. Cavanaugh, X. Sun, P. Liiva, D. Mao, J. C. Smith, and A. E. Bartels. Initial observations from the lunar orbiter laser altimeter (lola). *Geophysical Research Letters*, 37, 2010.
- John P. Snyder. *Map Projections: A Working Manual*. Geological Survey (U.S.), 1987. Professional Paper:1395.
- StreamComputing. Opencl vs cuda misconceptions. <http://streamcomputing.eu/blog/2011-06-22/opencl-vs-cuda-misconceptions>, Accessed: 17 September 2013. URL <http://streamcomputing.eu/blog/2011-06-22/opencl-vs-cuda-misconceptions/>.
- USGS. Map projections. <http://egsc.usgs.gov/isb//pubs/MapProjections/projections.html>, Accessed: 24 october 2014. URL <http://egsc.usgs.gov/isb//pubs/MapProjections/projections.html>.