Mars Craters Research Report

|  |  |  |
| --- | --- | --- |
| Sławomir Figiel *Space and Satellite Technologies, ETI Faculty Gdańsk University of Technology* [fivitti@gmail.com](mailto:fivitti@gmail.com) index no. 12345 | Tomasz Mrugalski *Space and Satellite Technologies, ETI Faculty Gdańsk University of Technology* [tomasz.mrugalski@eti.pg.edu.pl](mailto:tomasz.mrugalski@eti.pg.edu.pl) index no. 75751 | Ewelina Omernik  *Space and Satellite Technologies, ETI Faculty Gdańsk University of Technology* [ewelina.omernik@gm](mailto:ewelina.omernik@gm)ail.com index no. 12345 |

*Abstract* — The following report provides an overview of the research being conducted within the Mars Craters Research as part of the Remote Earth Observation Sensing class, held at ETI Faculty of Gdańsk University of Technology. This report covers an overview of the craters phenomena, their origins and creation process. The primary goal of the project was to identify satellite photos of Mars surface and identify craters on them. This report provides description of the methods used. A discussion about strong and weak points of various approaches applied conclude this paper.

Keywords —mars, craters, satellite, observation, opencv.

# Introduction

Mars is the most similar planet to Earth in our Solar System. Its sidereal day is 24h 37min, which is roughly 40 minutes longer than Earth’s. Mars gravity is weaker and equals to 37% of Earth’s gravity. Its atmosphere is probably the most distinct feature that sets Mars and Earth apart. Its average density (11.55 hPa) is around 0.6% of Earth’s (101.3 hPa). The atmosphere consists of 96% of carbon dioxide, 1.93% of argon and 1.89% of nitrogen. Oxygen is present only in trace amounts (0.174%). For more details, see [2].

Asteroids occupy various orbits throughout the whole Solar System, but the area behind Mars orbit has the highest concentration. Therefore it often nicknamed “an asteroid belt”. As of Nov. 2019 there are over 851.000 minor bodies known in the Solar System [3]. Earth is naturally protected from incoming minor bodies by its thick atmosphere and a large Moon that can alter the trajectory of incoming objects. Neither of those mechanisms as present on Mars. Also, the close proximity to main asteroid belt contributes significantly to the amount of surface impacts. As such, Mars is a very attractive place for meteorite research.

The goal of the research conducted was to study evidence of incoming bodies. Since the Mars’ atmosphere is so thin, most of the incoming meteors end up hitting the surface and becoming a meteorite. The primary goal of this research was to study the data available, experiment with various techniques and come up with the most reliable method for detecting craters.

# Methodology

There are currently 6 orbiters providing orbital photos from Mars: 2001 Mars Odyssey (2001), Mars Express (2003), Mars Reconnaissance Orbiter (2006), Mars Orbiter Mission (2014), MAVEN (2014) and ExoMars orbiter (2016). Year of reaching Mars orbit specified in parentheses.

Several imagers are particularly useful:

**THEMIS -** precision better than 10m on India territory and 20m on surrounding sea.

HiRISE – describe HiRISE here.

# Results

## Initial data selection

During our research we noticed that some impact craters have very bright edges on images from infrared sensor at night from THEMIS mission. NASA provide the global mosaic from these images in with a resolution of 100m. Several experiments proved that it’s a good source data for craters recognition. Using this data has one limitation, however. Not every crater has a bright edge. Crater must to have a specific minimal depth (at least 0.4-0.5 km) and the radius cannot be too big (< 15km).

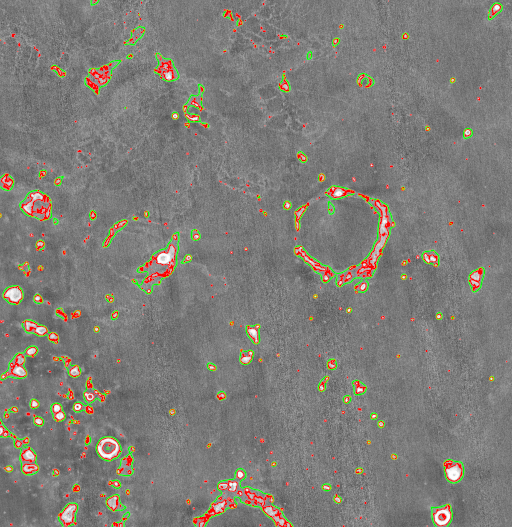
Another difficulty was introduced with an observation that craters are not the only objects that have a bright edges. Other terrain forms, such as hills, may show similar behavior. Fortunately, we came up with a way to filter them out.

## Early experiments

Initial experiments were conducted with GIMP [4], a general purpose image editing software with powerful plugins collection. The image selected for this experiment was coming from THEMIS and had a resolution of 300m. Two methods were investigated. The first method uses subtle Gauss blur, thresholding (with pixels within range 160-170 treated as white), dilate filter [5] that widened and enlarged the detected areas. Finally, the resulting white pixels were used as a mask. An example result of this method has been presented in Fig. 1, denoted red.

The second, alternative approach assumed somewhat similar approach, albeit the details were different. The first step used aggressive Gaussian blur, thresholding at the value 160, detecting edges using Gaussian differences and the second thresholding. The resulting white pixels were used as a mask. An example result of this technique is depicted in Fig. 1, denoted in green.

Several conclusions emerged from those experiments. Initial blur is necessary to remove minor noise distributed throughout the whole image. The blur parameters must be adjusted to the size of the craters being searched. This implies that to find craters of any sizes, several passes of the algorithm are necessary. Too aggressive blur performs poorly, even for large craters. They’re being merged with smaller craters in the area, resulting in unusable blobs of pixels. Small hills can distort the results. With increasing latitude, craters’ circumference become less pronounced and hills becoming more visible, thus making detection more difficult.



*Fig. 1: GIMP experiment*

# Pattern recognition using OpenCV

The next step in our research was to use OpenCV library [6]. OpenCV is an open source computer vision and machine learning software library. To ease the development, we used the basic python bindings for OpenCV [7] as well as extra wrapper for it [8]. Several experiments were conducted.

Every OpenCV based experiment had some common characteristic. First, it used the cv2 wrapper, loaded the input image file, processed the image with the results both being displayed on the screen and written to file. JPEG standard was used. In some cases the detected craters could be saved as a .CSV file and the detected shapes exported to .SHP file.

In all cases the primary mechanism for detecting craters was Hough algorithm for detecting circles.

## OpenCV experiment 1

In filtration stage we based on shape. We assume that impact crater have rounded shape. We use a circle recognition algorithm (for example Hough Circle Transform with Hough Gradient detection method).

During filtration you should remember that not whole edge will be bright for each crater. For old crater edge may be incomplete or edge may to have different structure and some parts are dark.

# Strong and weak points

This parameters gives 65% probability of the actual location to be within a circle of MRSE meters diameter. A slightly more practical estimate is 2DRMS, defined as doubles DRMS:

There are various versions of DOP parameters specified: GDOP (geometric DOP), PDOP (Position DOP), HDOP (Horizontal DOP), VDOP (vertical DOP).

All of those parameters can be tied together using the following equation:

# Definining obstuctions

Real world measurements are often affected by obstacles that prevent line of sight between a satellite and an observer. Trimble Planning software has a mechanism to define such obstacles. It allows the user to define direction of an obstacle and its height. Unfortunately, the interface provided is very imprecise and relies on accurate mouse movements. This approach is very error prone. Fortunately, it’s possible to write the obstacles definition to a file, edit it as needed and then load the data back.

The file syntax is very simple plain text file. Each line has an integer (expressing azimuth in degrees, between 0 and 359) followed by a floating point number (expressing height of an obstacle in said direction, expressed in degrees; allowed values are 0..90). Lines that are empty or start with semicolon are ignored.

The laboratory instruction specified the obstacles to be used as Az: ; Az: and . The author of this report developed simple editing tool called obstacles-edit that is able to generate and edit obstacles files. The tool requires python. It takes two arguments. The first one denotes a name of a file. The second one specifies min-max,h parameters that define starting and ending of an obstacle, h is obstacle height. An example execution of the script looks as follows:

python obstacles-edit.py obs.txt 160-250,60

python obstacles-edit.py obs.txt 300-355,70

python obstacles-edit.py obs.txt 45-50,55

After loading the file into Trimble Planning, the obstacles were visualized as presented on Fig.2. The software has been released under GPL v3 license and is available in [1].

# Conclusions

The inverse correlation between number of visible satellites and the GNSS accuracy has been clearly proven. Furthermore, another often neglected aspect – the satellites geometry on the sky – has been studied and was determined to have significant effect. Another important factor is the sky visibility and obstacles that can obstruct the line of sight. This problem, however, can be alleviated to a large degree by using more satellites.

##### Bibliography

1. S. Figiel, T.Mrugalski, E.Omernik “mars-obs”, open source project, <https://gitlab.klub.com.pl;3000/astro/mars-obs>, retrieved Nov. 2019
2. Wikimedia Foundation, Wikipedia page on Atmosphere of Mars, https://en.wikipedia.org/wiki/Atmosphere\_of\_Mars , Wikimedia, retrieved Nov 2019.
3. Minor Planet Center, <https://minorplanetcenter.net/about>,, Smithsonian Astrophysical Observatory, retrieved Nov. 2019
4. GIMP – GNU Image Manipulation Program, project website, <https://gimp.org>, retrieved Nov 2019
5. Dilate filter, GIMP Documentation, [https://docs.gimp.org/2.10/en/ plug-in-dilate.html](https://docs.gimp.org/2.10/en/%20plug-in-dilate.html), retrieved Nov. 2019
6. OpenCV project, <https://opencv.org/>, retrieved Nov 2019
7. OpenCV on Wheels, unofficial pre-built OpenCV package for Python, <https://pypi.org/project/opencv-python/>, retrieved Nov 2019
8. Simple wrapper for opencv-python, <https://pypi.org/project/opencv-wrapper/>, retrieved Nov 2019