

# 30330 IMAGE ANALYSIS WITH MICROCOMPUTER

PROJECT REPORT

by

KATLEEN BLANCHET s150798  
TITOUAN BOULMIER s150810

$$f(x+\Delta x) = \sum_{i=0}^{\infty} \frac{(\Delta x)^i}{i!} f^{(i)}(x)$$

$\Delta \int_a^b \Theta^{\sqrt{17}} + \Omega \int \delta \sqrt{e^{i\pi}} = -1$

$\infty = \{2.7182818284\}^{\circ}$

$\chi^2, \sum, \approx, \lambda$

!

December 17, 2015



Technical University of Denmark  
Department of Electrical Engineering

## Abstract

In order to solve the overpopulation issue, scientists are studying Mars features to make it habitable. In this context, a system capable of stabilizing, in real time, the rover arm in front of a Martian rock to allow researchers to study it could be helpful. This project intends to design such a system, through the realization of a depth mapping of the rock thanks to the structured light technique. First, a scene analysis is carried out to determine the camera and the needed aperture to get enough light and to choose an artificial light source capable of outshining the sunlight. To detect the beam spots projected as a unicolor grid by the green LED selected as additional light source, a color detection algorithm is performed in C/C++ with OpenCV, as well as a centroiding algorithm which takes into account the intensity of the pixels belonging to the points. Then, based on the triangulation principle, the distance camera-target is derived and implemented.

The system is verified, with an alternative equipment due to lack of time, through two indoor experiments with one laser beam point and a line of dots. The results are quite promising, since the errors between the calculated distance and the real one could be reduced with an improvement of the calibration and of the color detection. The real time computation is validated. An outdoor experiment has also confirmed that the artificial light source should be capable of outshining the sun on the Red Planet. Nevertheless, extra work is required to implement the real system and the use of patterns, other than a unicolor grid which cannot prevent patterns disorders, could permit to get a more accurate 3D map.

---

## Acknowledgments

We would like to express our gratitude to our supervisor Prof. Alessandro S. Massaro for his advices throughout the duration of the project.

We would also like to thank Prof. John Leif Jørgensen for his availability when we had additional questions.

Beside them, we thank the associate professors to the course who taught us the knowledge needed to carry out this project.

# Contents

<b>Introduction</b>	<b>1</b>
Context	1
Problem formulation	2
Problem delimitation	3
<b>1 Theory Section</b>	<b>5</b>
1.1 Mars Features	5
1.1.1 Atmosphere of Mars	5
1.1.2 Climate of Mars	6
1.1.3 Albedo	7
1.2 State of Art of the Depth Mapping	7
1.2.1 Principal Techniques Based on Image Analysis	8
1.2.2 M-array Pattern Projection Method [10]	8
1.2.2.1 Pattern Disorders	9
1.2.2.2 Correction of Pattern Disorders	9
1.2.3 Grid Indexing	12
1.2.3.1 Mini-patterns Used as Code Words	12
1.2.3.2 2D Array of Color-coded Dots	12
1.3 Image processing	12
1.3.1 HSV Color System	13
1.3.2 Morphological Operations [3]	14
1.3.2.1 Erosion	14
1.3.2.2 Dilation	15
1.3.2.3 Opening	15
<b>2 Development</b>	<b>16</b>
2.1 Scene Analysis	16
2.1.1 Camera	16
2.1.1.1 CCD	16
2.1.1.2 Field of View	17
2.1.1.3 Focal Length	17

2.1.1.4	Aperture . . . . .	18
2.1.2	Artificial Light Source . . . . .	19
2.1.2.1	Design of the Light Source . . . . .	19
2.1.2.2	Validation of the Light Source . . . . .	21
2.1.3	Signal Noise Ratio . . . . .	22
2.1.3.1	Irradiance . . . . .	22
2.1.3.2	Target's Radiance . . . . .	23
2.1.3.3	Target's Irradiance . . . . .	23
2.1.3.4	Light Source Irradiance . . . . .	24
2.1.3.5	Signal/Noise Ratio . . . . .	25
2.1.4	Summary and Comparison with MER Cameras . . . . .	28
2.1.5	Cost and Power Consumption . . . . .	29
2.2	Distance Camera-Target of One Point . . . . .	30
2.3	3D Map . . . . .	33
2.4	Image Processing . . . . .	34
2.4.1	Color Detection . . . . .	34
2.4.2	Beams Distinction . . . . .	36
2.4.3	Centroiding . . . . .	38
<b>3</b>	<b>Experimental Results</b> . . . . .	<b>41</b>
3.1	Equipment Employed for the Experiments . . . . .	41
3.1.1	Lab Webcam Philips SPZ2000 . . . . .	41
3.1.2	Laser Pointer . . . . .	42
3.1.3	Computer . . . . .	42
3.2	Parameters of the Webcam . . . . .	42
3.2.1	Determination of the Focal Length . . . . .	42
3.2.1.1	Triangulation Method . . . . .	42
3.2.1.2	Camera Calibration . . . . .	43
3.2.2	Discussion About the Lens Distortion . . . . .	44
3.3	Determination of the Depth of One Laser Beam . . . . .	45
3.3.1	General Procedure . . . . .	45
3.3.1.1	Calibration . . . . .	45
3.3.1.2	Distance Computation . . . . .	46
3.3.2	Experiment and Results . . . . .	46
3.4	Distances with Several Points . . . . .	49
3.4.1	Choices of Implementation . . . . .	49
3.4.2	Progress of the Experiment . . . . .	49
3.4.3	Results . . . . .	51
3.4.4	Interpretations . . . . .	52

3.4.5	Limits of the Experiment . . . . .	54
3.5	Outdoor Integration Testing . . . . .	54
<b>Conclusion</b>		<b>56</b>
<b>Future Work</b>		<b>58</b>
<b>Appendices</b>		<b>60</b>
.1	Results of the Experiment With a Line of Dots . . . . .	60
.2	LED Properties . . . . .	62
.3	Matlab Camera Calibrator Toolbox . . . . .	63
.4	CCD Properties . . . . .	64
<b>Bibliography</b>		<b>84</b>

# Introduction

## Context

The research of habitable planet is one of the major concerns of human beings. Indeed, this discovery, in addition to giving hope of locating another species, would bring solutions to overpopulation. According to scientists, the main criterion for life is water. Since the properties of celestial bodies are disparate, a habitable zone has been defined, considering that water can only exist at a specific range of temperature. If the temperature of the planet is too low, the water will freeze and conversely, it will evaporate if the temperature is too warm. The habitable zone is not fixed; it moves with the evolution of the sun. Mars is thought to have belonged to the zone once, due to its most hospitable climate after Earth. For a matter of fact, NASA's Mars Reconnaissance Orbiter has recently provided a strong evidence of water currently flowing on the Red Planet. It may then be possible to live on Mars and that is the aim of Mars one project: send a colony over there. The terraforming, which would change characteristics of Mars to make it suitable for humans, is also worth considering before colonization. In order to do so, scientists need to learn more about the features of the planet. Orbiters and rovers are already scanning mars surface and soil. Nevertheless, they lack the depth information, only equipped of Two Dimensional (2D) cameras. Their three dimensional (3D) scans have permitted to acquire it and to print a Martian meteorite on Earth in 2014. This information is useful for a better understanding of the rocks properties but that is not enough. Some missing data have delayed the 3D print. It could be interesting to provide to scientists a real time 3D map of a selected rock. It would be easier to study stones but the map could also help the robot to stabilize. With wind and an uneven ground, the rover is liable to move. The pattern recognition of a rock would be more robust by adding a 3D map. Samples could be collected being sure that the robot has taken the right piece.

Furthermore, the 3D map could have several other applications. The features of the Martian soil acquired by the orbiting satellites are not exhaustive and lack of accuracy. Thus, applied to the area in front of the robot, the 3D map could detect new obstacles and update in real time the available map to distinguish modifications of the ground. It could also permit spacecrafts to determine a flat area to land in real time by analyzing the surface of the planet from space and during the descent. Moreover, the 3D map can be applied in completely different fields. Indeed, this technique can be found, for example, in video games with environment recordings for real time augmented reality gaming, in medicine with skin surface or skin roughness measurement, in cosmetic with wrinkle measurement and also in industry with 3D-automated optical inspection, volume measurement or classification of grinding materials and tools.

However, the 3D mapping is not very common and is still improving. In this context, designing a system capable of carrying out a 3D map of a Martian rock would be

profitable for future research. This is the purpose of this project, undertaken during the Image Analysis with Microcomputer course, under the supervision of Alessandro Massaro. Combining image and scene analysis, programming and robotics seems relevant for learning to work as an engineer.

This report is intended to our fellow students. The first part will bring additional knowledge required to understand the next developments. The design of the system is described in the next section, followed by its verification. To end, future work is addressed.

## Problem formulation



Figure 1: Curiosity Rover with an embedded camera on a arm [11]

In this report, we assume that we would like to send a rover on Mars, capable of communicating with Earth. Self-powered by solar panels, it will land at latitude  $50^{\circ}$  where it could get enough sunlight to recharge its lithium batteries. It is provided with an arm whose hand is replaced with a camera (see figure 1). The latter will be used by scientists to observe relevant rocks to study. In order to accomplish this mission, once a stone is designed, the camera must be able to keep it in focus, despite the wind, the movement of the robot or any other perturbation. This implies a real time image analysis to be able to rectify the position of the camera. As a robust method is needed to maintain the stone in front of the arm, the correspondence problem is achieved by carrying out a 3D map of the surface. A luminous source should then be added to the rover to be projected on the surface containing the rock and detected by the camera.

This luminous source must be powerful enough to outshine the sunlight during the day, notwithstanding that the energy needed to make it work has to be negligible compared to the amount provided to the rover. Moreover, the characteristics of the camera need to be

perfectly adapted to Mars, as once the rover has landed on the red planet, it would be impossible to adjust it.

Will it be feasible to design such an embedded system, composed of a camera, a luminous source and algorithms, capable of keeping a rock in focus thanks to a 3D map, for an application on Mars soil?

## **Problem delimitation**

First of all, we assume that the rover is already in front of the rock, that is to say, we do not take into account the path to go close to the target with the different potential obstacles that may be present on the way. Then, in order to design the rover's camera which will be used to study rocks and to implement a robust algorithm to carry out a 3D map of the surface of the rock, different characteristics of Mars and of the target need to be specified. However, as all of them cannot be taken into account, some simplifications and choices will be made.

### **Mars delimitation**

Plenty of missions on Mars have been achieved and a great deal of data has already been gathered. Nevertheless, even if some of them will be used to design our system, others will be simplified or even ignored. The first simplification concerns the atmosphere of the Red Planet. Indeed, even if its composition is now well known, it will be assumed that the dirt on the Mars surface plus the different layers of the atmosphere absorb, or scatter, 10% of the solar energy. Moreover, the influence on the image acquisition that the dirt between the target and the camera could have will not be taken into account. The second reduction covers the temperature. Indeed, even if it can reach -143°C during winter, 27°C during summer and have around 60°C variations between daytime and nighttime [13], we will suppose that the CCD sensor works well all the time.

### **Target delimitation**

Regarding the target, that is to say the part of the rock being studied, it is supposed to :

- be vertical;
- not exceed 2\*2 meters;
- have an area between 0.1 and 1 square meters;
- have a relief less than 0.5 meters.

### **Camera delimitation**

Then, regarding the camera which is designed during this study, it is presumed to :

- be between one and two meters from the target;
- be right in front of the target, that is to say that the angle between the normal of the target's surface and the focal axis of the camera is  $0^\circ$ ;
- be able to capture the image of a target of 2 meters height maximum.

The different characteristics of the target and the camera are represented figure 2.

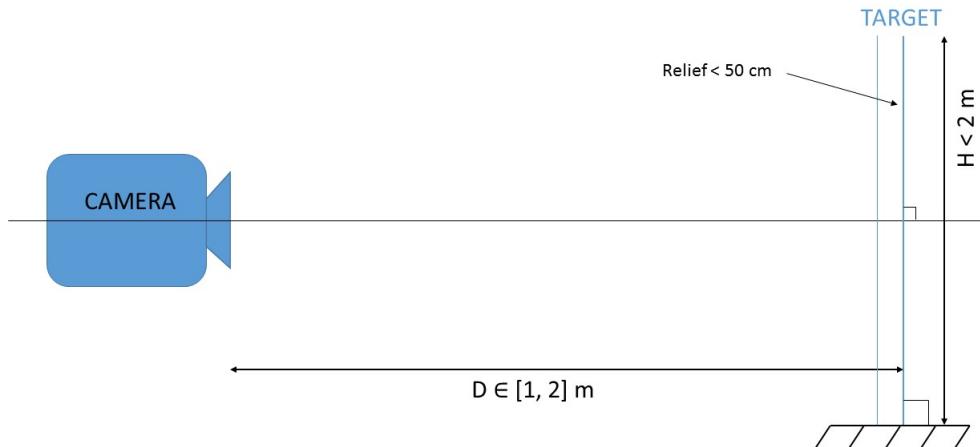


Figure 2: Schema of the scene

It is also supposed that as there is a limited amount of power, only 300 mW are available for the artificial light source. Finally, the last assumption is that the wavelength taken into account is from 400 to 800 nm. Indeed, the first reason is that the CCD sensor chosen (see part 2.1.1.1) works with this range of wavelength. Moreover, this span corresponds almost to the visible spectrum (380 - 750 nm), with which the 3D algorithm works, and as Mars has a reddish color and a lot of infrared, the spectrum is expanded to 800 nm.

# Part 1

## Theory Section

### 1.1 Mars Features

Mars, the fourth planet from the Sun (1.3814 to 1.6660 AU) is the second smallest planet of the the Solar System (its radius is 3389.5 km). Known as the “Red Planet” because of the amount of iron oxide on its surface, it has an orbital period of 668.6 sols (Mars’ solar day), that is to say 687 days, with an average day length of 24h37m.

#### 1.1.1 Atmosphere of Mars

Because of its thin atmosphere, Mars has surface features which present analogies with both Moon, through the impact craters, and Earth, with volcanoes, deserts, polar ice caps. It is mainly composed of carbon dioxide  $CO_2$  ( $96,0 \% \pm 0,7 \%$ ), argon  $Ar$  ( $1,93 \% \pm 0,01 \%$ ) and oxygen  $O_2$  ( $0,145 \% \pm 0,009 \%$ ). As the gravity of Mars is low, the height of the atmosphere is 11 km, more than one and a half times higher than the Earth’s one (7 km). Moreover, due to this low gravity, the wind can disrupt a rover easier than on Earth and a large amount of suspended dust is perpetually present at the surface.



Figure 1.1: Dust Devil. Mars Reconnaissance Orbiter, made by HiRISE on Feb. 16, 2012 [2]

Regarding the dust, of which the particles have a mean diameter of  $1,5\mu\text{m}$ , it is continually injected into the atmosphere by wind or dust devils (a 12 km high one can be seen on Figure 1.1). Such eddies are far from being anecdotal because they bring dust into significant volumes of the atmosphere. Even if the amount of dust is never very massive, it can be lifted by wind whose speed is slower than  $2\text{ m/s}$  and maintained indefinitely with a wind force of only  $0,8\text{ m/s}$ .

Moreover, regarding the wind on Mars, in low altitudes, the Hadley circulation<sup>1</sup> dominates and its principle is similar to the one which, on Earth, generates the Trade winds. In high altitudes, a battery of areas of high and low pressure, named baroclinic pressure waves<sup>2</sup>, controls the weather. The Mars wind lifts a large amount of dust and can even trigger huge dust storms which can affect the whole atmosphere (see Figure 1.2). Cyclonic storms can also appear on Mars. Indeed, cyclones comparable to the ones on Earth tend to create during summer in the northern hemisphere but only in high latitudes.



Figure 1.2: Two views of Mars with spatial telescope Hubble before and after the big dust storm in 2001 [1]

### 1.1.2 Climate of Mars

Because of its distance to the Sun bigger than the Earth one, Mars receives less solar energy 2.1.3.1. Additionally, as its atmosphere is thinner, there is only a negligible greenhouse effect, and hence the average temperature is around  $-63^\circ\text{C}$  on the surface of Mars with wide variations between daylight and night.

Then, the obliquity of Mars is close to the Earth one (respectively  $25,19^\circ$  and  $23,44^\circ$ ) but the eccentricity of Mars is bigger (0,09332 against 0,01671 for Earth) which means that, even if the Red Planet has similar seasons to Earth, they have different intensity and

<sup>1</sup>movement on a planetary level of the layers of gases surrounding the planet

<sup>2</sup>In meteorology a baroclinic atmosphere is one for which the density depends on both the temperature and the pressure

duration. Thus, the northern hemisphere has seasons less pronounced than the southern hemisphere because its aphelion is at the end of spring and its perihelion at the end of autumn. Thus, there are short and soft winters and long and fresh summers in the northern hemisphere. On the opposite, the southern hemisphere has very marked seasons with long and cold winters and shorter and warmer summers than in the northern hemisphere. That is why there are higher temperature differences in the south.

### 1.1.3 Albedo

Mars surface is covered by sand and volcanic rocks. The first purpose of this project is to allow scientists to examine these rocks through the camera. To achieve it, it is needed to know their characteristics, especially their albedo. As a reminder, the albedo is the fraction of incident light which is reflected from a surface. We can assume that the power reflection of Martian rocks is the same than on Earth. Then, to cover a wide range of rocks, the albedos of a black and of a white stone will be considered. Charcoal, as a dark rock, is a powerful absorber of the sun radiation, with an albedo around 0.05. On the contrary, chalks are poor absorbers and their albedo reach 0.45 according to [7]. In the following parts of the report, it will be taken for granted that Martian rocks have an albedo between 5 and 45%.

## 1.2 State of Art of the Depth Mapping

Traditional cameras only take 2D pictures, with horizontal and vertical information. To reproduce the human vision useful in various applications such as the stabilization of a Martian rover arm in front of a rock to be analysed, it is often needed to add the depth data to get 3D coordinates of each pixel of the images. The depth mapping, also called 3D surface measurement or 3D imaging techniques etc, is carried out by two non-contact major methods as specified by [12]: “projecting (in the active form) or acquiring (in the passive form) electromagnetic energy onto/from an object followed by recording the transmitted or reflected energy”. They can be divided into two categories, the non-optical and optical sensing. The first one generally uses acoustic and electromagnetic sensors to determine the distance from the system to the object, by assessing the duration of a round trip of a pulse. In the latter the light permits to get the depth. We will focus on this last technique as it will be more convenient to achieve it on Mars. Active optical sensing uses an additional light source where as passive optical sensing works only with the irradiance reaching the scene and the radiance of the object.

### 1.2.1 Principal Techniques Based on Image Analysis

We will address only methods using image analysis as this is the purpose of the course. Stereo vision and Photogrammetry systems are passive. The principle of the first is based on the utilisation of two or more cameras to record the scene. The reconstruction of the 3D is carried out by solving a correspondence problem with the identification of patterns in two or several images. The photogrammetry method calculates the depth by taking pictures from different points of view of a scene, through a camera preliminary calibrated. Both techniques demand a high computing power. Thus they are dismissed since the Rover cannot provide too much power to the embedded processor.

In active optical sensing, laser triangulators and structured light can be distinguished. They share the same approach: a beam or a pattern is projected through a laser source towards the object and its position on the image acquired by the camera is measured. Then, the distance from the camera to the object (the depth) is found by applying a triangulation technique. The difference between the two methods is that it is needed to scan the whole object for the laser triangulators one. Indeed, only a beam or a laser strip reaches it at once. The experiment has to be repeated several times to have an exhaustive 3D map. Regarding the structured light approach, unique patterns are projected simultaneously which allows to obtain the depth data at once if the object is small but in any case, it reduces a lot the number of acquisitions compared to the laser triangulators. According to [4], the patterns can be in 3D but they are rare. 2D patterns are the most common and easier to implement. Depending on the scene motion, the structured light method can be classified into multi-shot and single-shot. If the scene is moving, the acquisition time must be short and thus, one picture should be enough to get the desired information. On the other hand, if no constraint is given, several pictures can be taken and a pattern adapted to sequential can be selected. In our case, the scene does not move but the robot arm could move and thus needs to be stabilized. That is why only single-shot structured light will be considered.

In the future part of the report, the implementation and experiments will be performed through laser triangulators techniques and structured light with a unicolor grid as a pattern. They are easier to set up but will only give us a partial depth map. To obtain a complete 3D map in a single shot, more complex patterns should be applied to the martian rock. Some methods that could be carried out then are described below.

### 1.2.2 M-array Pattern Projection Method [10]

The binary M-array pattern projection method consists in projecting dark and light dots. The color of each dot represents a 0 (dark) or 1 (light) of the M-array pattern. The spatial coordinates of each observed dot matched with one of the initial pattern can be determined by triangulation. However, pattern disorders occur often in the observed pattern and some

dots cannot be matched. One of the asset of this method is the detection and correction of pattern disorders.

### 1.2.2.1 Pattern Disorders

According to [10], there are three different types of pattern disorder.

- Loss of dots: If there are several objects in the scene, an object may obstruct the pattern. The first, second and third dots on the figure 1.3(a) are not visible by the camera on figure 1.3(b).
- Displacement of dots: in comparison to the initial pattern projected, the dots may shift because of the depth of each object. On the figure 1.3(a), the dots 4 to 6 on the object 2 have shifted to the left compared with the dots of object 1.
- Interchange of dots: if an object is in front of another, the order of dots may change. The third dot on figure 1.3(a) has changed of order and is the fourth from the point of view of the camera (figure 1.3(b)).

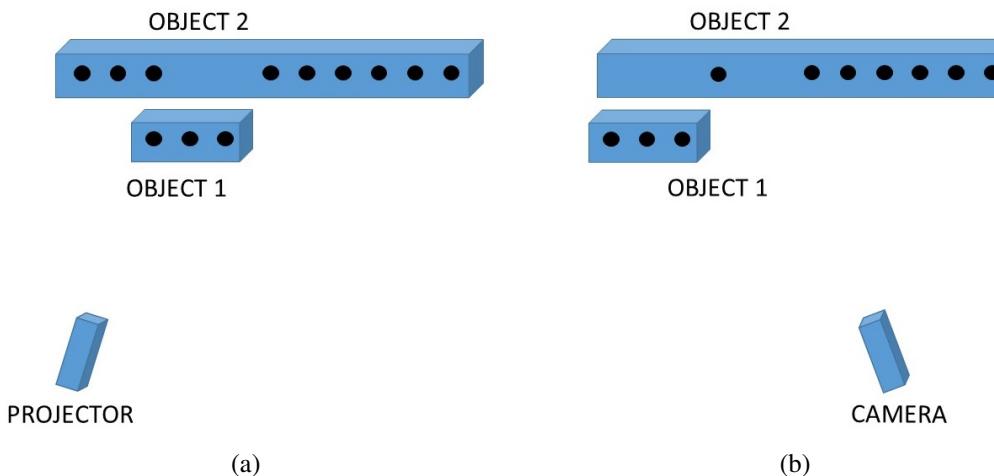


Figure 1.3: Pattern Disorders

### 1.2.2.2 Correction of Pattern Disorders

There are four steps in the matching of the observed dots with those of the pattern and the correction of the pattern disorders.

#### Temporary Array

The image recorded is analyzed row by row of dots and a 2-D array which contains the numbers of the observed dots is created. This array will be used to differentiate each dot. The array can be computed by these steps :

- Creation of the array (the elements are initialized to zero).
- Scan of the image to detect dots. Each time a point is detected, its value and location (center) are saved into 1-D arrays VALUE, X and Y.
- The position of the dots are quantized according to the initial pattern using X and Y so that adjoining dots have successive quantized coordinates.
- Insertion of the number of each dot into the array (the location into the array corresponds to the quantized coordinates).

This temporary array (T-array) may have some vacant places due to disorder pattern.

### **Grow**

Each element of the T-array gets an index (column number of the M-array). To do so, a window is defined and applied to the T-array and the M-array where the elements into the window of each array have the same value. The elements into the window of the T-array can be indexed using the M-array and the two windows moved simultaneously by one column or row. This is repeated until a difference between the two windows is detected. New windows are then initialized elsewhere. The grow terminates when no window can be initialized to an area of the T-array where there is no index yet.

### **Detection of Pattern Disorders**

The detection step is carried out during the growing and is explained below using an example. We use a  $3 \times 4$  window.

- step 1: the region growing starts at  $(0', 0')$  and  $(0, 0)$  on the T-array and M-array (see Figure 1.4 a).
- step 2: the window slides to the right until a difference between the two arrays is detected (see Figure 1.4 b).
- step 3: In the T-array, the window is shifted to the right until the error is on the left side of the window. In the M-array, the window is displaced until it matches with the one of the T-array (see Figure 1.4 c).
- step 4: the windows now slides to the left and all the indices of the columns of the T-array are readjusted according to the two windows until another difference is detected.

Then, as regions of the T-array extend because of the step 4, a new window may be initialized on the extended region and the growing starts once more. At the end, except some array cells corresponding to dots lighted near the border of two objects, all the indices of the T-array have been determined.

	0'	1'	2'	3'	4'	5'	6'	7'	8'	9'
1'	1	0	0	1	1	1	1	1	1	1
2'	0	0	1	0	1	0	0	1	0	0
3'	0	0	0	0	1	1	1	0	0	1

T-array

	0'	1'	2'	3'	4'	5'	6'	7'	8'	9'
1'	1	0	0	1	1	1	1	1	1	1
2'	0	0	1	0	0	1	0	1	0	0
3'	0	0	0	0	0	1	1	1	0	1

T-array

	0	1	2	3	4	5	6	7	8	9	10	11
1	1	0	0	1	1	1	1	1	1	1	1	1
2	0	0	1	0	1	0	1	0	0	1	0	0
3	0	0	0	0	1	1	1	1	0	0	1	

M-array

a

	0	1	2	3	4	5	6	7	8	9	10	11
1	1	0	0	1	1	1	1	1	1	1	1	1
2	0	0	1	0	0	1	0	1	0	0	1	0
3	0	0	0	0	0	1	1	1	0	0	1	

M-array

b

	0'	1'	2'	3'	4'	5'	6'	7'	8'	9'
1'	1	0	0	1	1	1	1	1	1	1
2'	0	0	1	0	1	0	0	1	0	0
3'	0	0	0	0	1	1	1	0	0	1

T-array

	0'	1'	2'	3'	4'	5'	6'	7'	8'	9'
1'	1	0	0	1	1	1	1	1	1	1
2'	0	0	1	0	0	1	0	1	0	0
3'	0	0	0	0	0	1	1	1	0	1

T-array

	0	1	2	3	4	5	6	7	8	9	10	11
1	1	0	0	1	1	1	1	1	1	1	1	1
2	0	0	1	0	1	0	1	0	0	1	0	0
3	0	0	0	0	1	1	1	1	0	0	1	

M-array

c

	0	1	2	3	4	5	6	7	8	9	10	11
1	1	0	0	1	1	1	1	1	1	1	1	1
2	0	0	1	0	0	1	0	1	0	0	1	0
3	0	0	0	0	0	1	1	1	0	0	1	

M-array

d

Figure 1.4: Detection of the disorders

### Correction of Pattern Disorders

Knowing that, in the T-array, the elements which still have no indice appear to be mostly alone or consecutive, some rules permit to correct the index of these elements.

- if the element above and the one below have the same index k, the current element has also this index k.
- if the element at the left has the index k-1 and the one at the right k+1, then the current element has the index k.
- the difference between the index of two horizontal neighbors corresponding to closer positions on the image is one.
- if the indices of a vertical sequence are still undetermined, they get the index of the element at the top or bottom of the vertical.
- if the indices of a consecutive horizontal sequence are still undetermined, they get the index such as it increases by one from index determined on the left to the one on the right. 

k-5	k-4	k-3	<b>k-2</b>	<b>k-1</b>	<b>k</b>	<b>k+1</b>	k+2	k+3	k+4	k+5
-----	-----	-----	------------	------------	----------	------------	-----	-----	-----	-----

According to [10], the previous rules should correct the most of the errors of indexation and the cases where the error cannot be corrected are scarce.

### 1.2.3 Grid Indexing

The method of the binary M-array projection is based on the matching of a binary array, but other patterns can be used too to permit the matching of the projected and observed patterns. A non-exhaustive list of grid indexing is presented below.

#### 1.2.3.1 Mini-patterns Used as Code Words

Instead of using a binary array, an array of words coded by patterns can be employed. Each pattern codes a word. As instance, on the figure 1.5, three words have been coded and permit to convert the projected pattern into a array of integers.

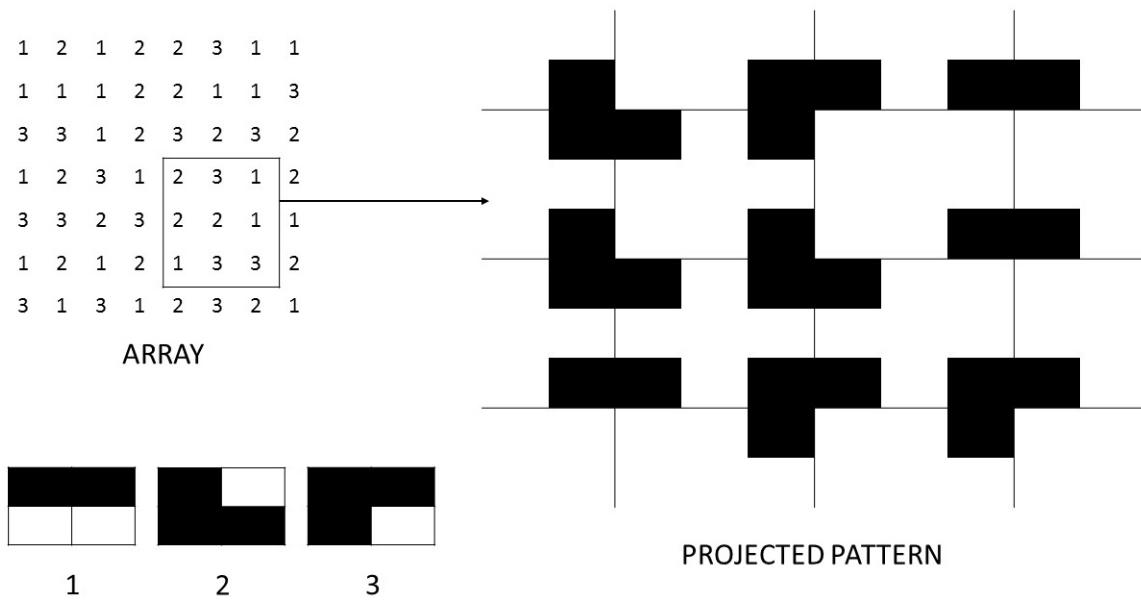


Figure 1.5: Mini-pattern used as code words

#### 1.2.3.2 2D Array of Color-coded Dots

An other technique is the 2D array of color-coded dots. The concept of this method is almost the same as the previous one but instead of using pattern to code a word, colors are manipulated (see Figure 1.6).

## 1.3 Image processing

In the following part of this report, the color detection will be used to distinguish color light points projected by a luminous source on a Martian rocks from its background. In order to implement this color detection, the HSV color model needs to be defined, as well as some morphological operations.

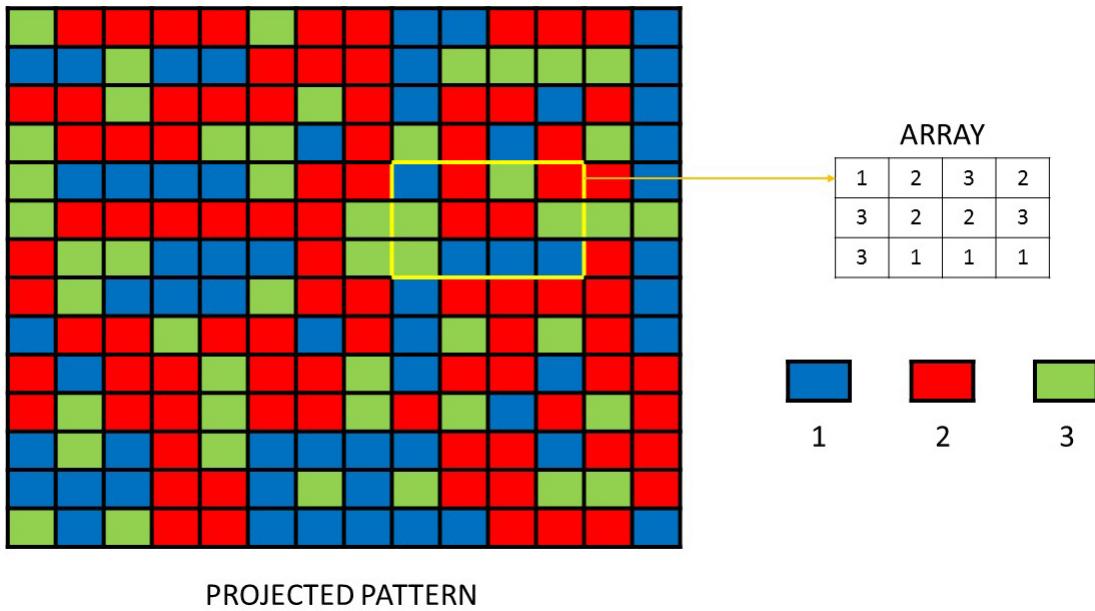


Figure 1.6: 2D Array of Color-coded Dots

### 1.3.1 HSV Color System

HSV stands for Hue Saturation Value. It is a color system which allows the separation of the color components from the intensity, unlike RGB system. It is more intuitive to find a plain object by varying HSV parameters.

The Hue designates the color itself. As it can be seen on the figure 1.7, the color is determined by an angle between  $0^\circ$  and  $360^\circ$ . Red corresponds to  $0^\circ$ , green to  $120^\circ$  and blue to  $240^\circ$ . OpenCV range is 0-180. Changing the Saturation will affect the intensity of the color. A low value indicates a dull color. The Value has an impact on the luminosity of the color. Decreasing this value makes the color darker. The S and V are varying between 0 and 1 but the OpenCV range is 0-255.

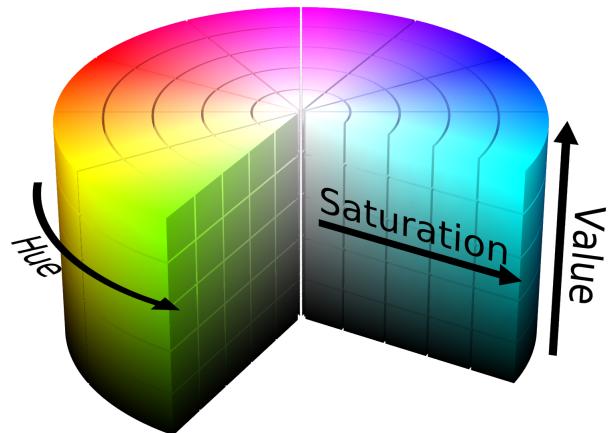


Figure 1.7: HSV diagram [15]

### 1.3.2 Morphological Operations [3]

The morphological operations on a binary image consists in applying a structuring element to each pixel of the object parts to reshape them. Two basic operations will be used later on and are explained here: the erosion and the dilation. Detailed diagram of the erosion and dilation can be found below (figures 1.8 and 1.9). It should be noticed that for more legibility, the background is in white and the object in color. It will however be assumed that the pixels of the object have a value superior to the background.

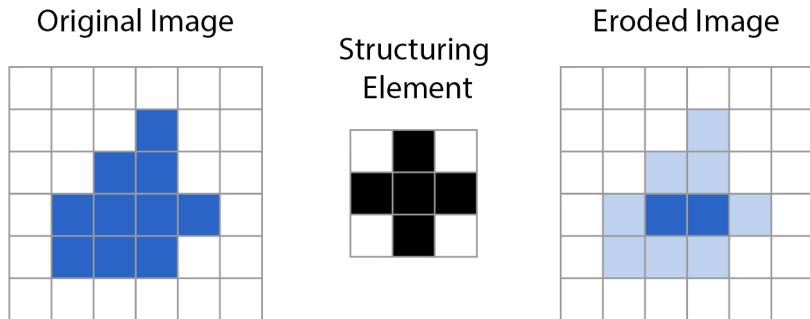


Figure 1.8: Erosion principle

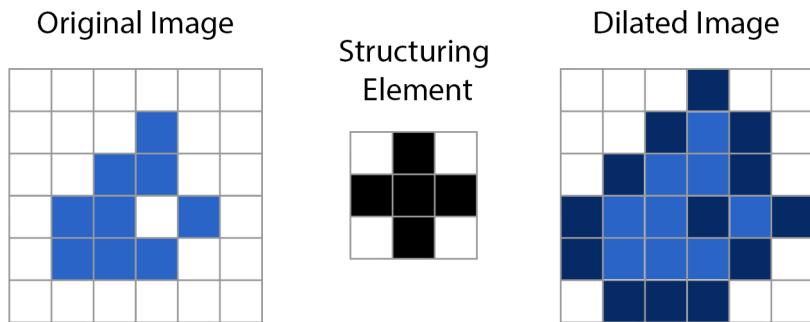


Figure 1.9: Dilation principle

#### 1.3.2.1 Erosion

This transformation is similar to the logic gate AND. The structuring element or kernel is moved along the image and the minimal pixel value is set as the new value of the anchor point, in the figure, the center of the structuring element. This means that if all the points of the kernel are on an object (a white object on a black background), the anchor point will keep its value. On the contrary, if any element of the kernel is out of the object, the center point will become part of the background. The erosion is useful to separate two objects connected and to remove objects which are smaller than the size of the structuring element (noise).

### **1.3.2.2 Dilation**

The dilation is the opposite transformation of the erosion. It corresponds to the OR. If one element of the kernel is on the object (which is still assumed to have the maximal value for its pixel), the anchor point will be set to this value and thus be turned into a part of the object. This operation permits to fill holes in an object, to increase its size and to link objects which are closer than the size of the structuring element.

### **1.3.2.3 Opening**

When combined, first by operating an erosion and then a dilation, the transformation is called an opening. The inverse operation is the closing. An opening could be convenient to separate luminous points which are connected and to then increase their size, lessen by the erosion to retrieve a shape similar to the original points.

# Part 2

## Development

### 2.1 Scene Analysis

Into this section, the camera and the artificial light source are designed and the Signal/Noise ratio is calculated.

#### 2.1.1 Camera

In order to design the camera and find its characteristics, a scene analysis have to be carried out. Our study is based on the MER (Mars Exploration Rover) cameras properties [8]. To begin with, we choose an image sensor.

##### 2.1.1.1 CCD

The Charge Coupled Device (CCD) is commonly more sensitive to light than its counterpart, the CMOS (Complementary Metal Oxide Semiconductor). Moreover, in the near infrared CCD appears to have a better response. Since Mars has a reddish color, it seems more accurate to select the CCD detector.

As scientists will need to discern the details of rocks, the resolution should be around the megapixel. Taking into account the cost, which will increase with the resolution, and the time of computation for an image with too many pixels, 1024 x 1024 seems to be an acceptable compromise.

Finally, the last element to consider is the pixel size. A trade-off have to be found between having a higher resolution (smaller pixels) and more sensitivity (larger pixels). All the camera of the MER mission were conceived with a pixel size of  $12 \times 12 \mu\text{m}^2$  for a resolution of 1024 x 1024. It was decided to comply with that value since it is quite a large pixel value which allows a better performance.

For the next parts of this report, we will use the data sheet of the FTT1010M CCD image sensor (see Appendix .4) as a basis for our calculations which will require CCD

details.

### 2.1.1.2 Field of View

According to the book [5], the Field of View (FoV) “is the angle of the cone of directions encompassed by the scene that is being images”. This solid angle is needed to compute the focal length the camera should have.

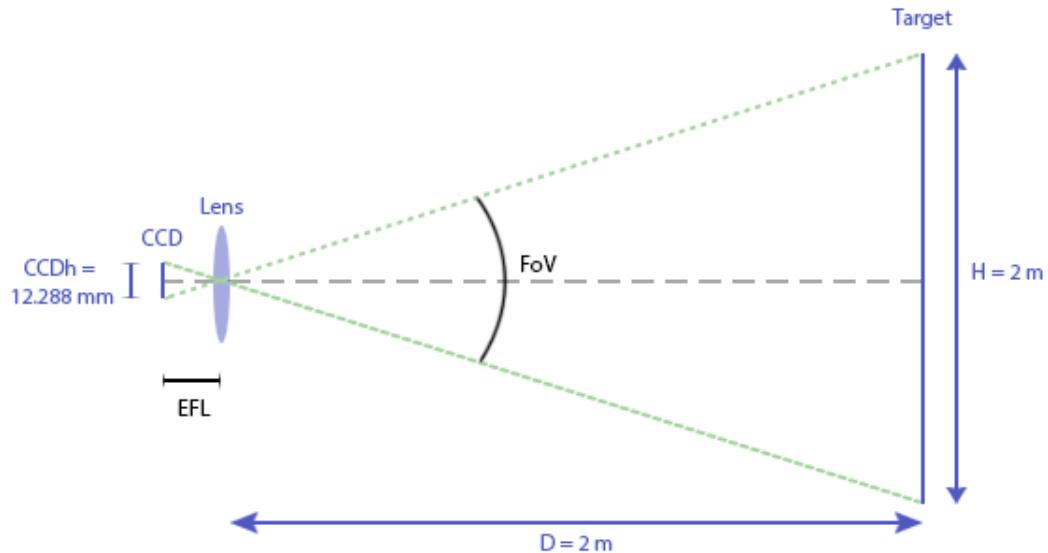


Figure 2.1: Field of View and Effective Focal Length

A simple relationship between the distance lens-target, the size of the target and the FoV can be deduced from the detailed diagram 2.1:

$$\tan\left(\frac{FoV}{2}\right) = \frac{1}{2} \times \frac{H}{D}$$

We derive and get

$$\frac{FoV}{2} = \arctan\left(\frac{1}{2} \times \frac{H}{D}\right)$$

Numerically, with our problem delimitation, the Field of View is equal to  $26.6^\circ \times 26.6^\circ$ .

### 2.1.1.3 Focal Length

Thanks to the previous diagram, the Effective Focal Length (EFL) can also be determined:

$$EFL = \frac{1}{2} \times \frac{CCDh}{\tan\left(\frac{FoV}{2}\right)} = 12.29 \text{ mm}$$

where  $CCDh$  is the image section of the CCD according to the data sheet.

In order to get the real focal length, we choose  $rf = 1 \text{ m}$  for the distance of focus of the camera.

$$f = \frac{1}{\frac{1}{rf} + \frac{1}{EFL}} = 12.14 \text{ mm}$$

#### 2.1.1.4 Aperture

To determine the diameter of the aperture, several criteria have to be accounted for. If the diameter is too small, the sharpness of the image will decrease due to the diffraction effect. In fact, at large aperture, the diffracted light is negligible compared to the total amount of light entering the system. On the other hand, we should also consider the Depth of Field (FoV). We need to insure that it is big enough to allow us to see the whole target on the image. The relationship between the DoF and the diameter is inversely proportional. That means that if we increase the diameter, we will lessen it. The Diameter of Confusion (DoC), linked to the DoF is also a factor to take into consideration for the choice of the diameter. According to Wikipedia [14] it corresponds to “an optical spot caused by a cone of light rays from a lens not coming to a perfect focus when imaging a point source”. The smaller the DoC is, which corresponds to a better focus, the bigger is the DoF, and also the smaller is the diameter.

As the behaviour of the depth of field and of the diameter of confusion runs counter to the diffraction one, a trade-off has to be found.

These formula are used to calculate the diameter of confusion and the diffraction spot:

$$DoC = Dsr \cdot \frac{|r - rf|}{r} \cdot \frac{f}{rf - f}$$

where  $Dsr$  is the diameter of the aperture

$$DiffractionSpot = 2 \cdot EFL \cdot \tan\left(1.22 \cdot \frac{\lambda}{Dsr}\right)$$

where  $\lambda$  is a wavelength of the sunlight

In the delimitations, it is assumed that the wavelengths of the sunlight belong to [400 800] nm. To settle on a diameter, the diameter of confusion and the diffraction spot were computed for diameters varying from 1 mm to 5 mm and wavelengths from 400 nm to 800 nm. The results are shown graphically in figure 2.2. As the diameter of confusion cannot be bigger than the height of a pixel, the line corresponding to  $12 \mu\text{m}$  was also added to the graphic. In order to have the best trade-off, that is to say all the curves under  $12 \mu\text{m}$ , we have chosen a effective lens entrance aperture  $Dsr$  equal to  $1.9 \text{ mm}$ , corresponding to

$DoC = 0.0117 \text{ mm}$ .

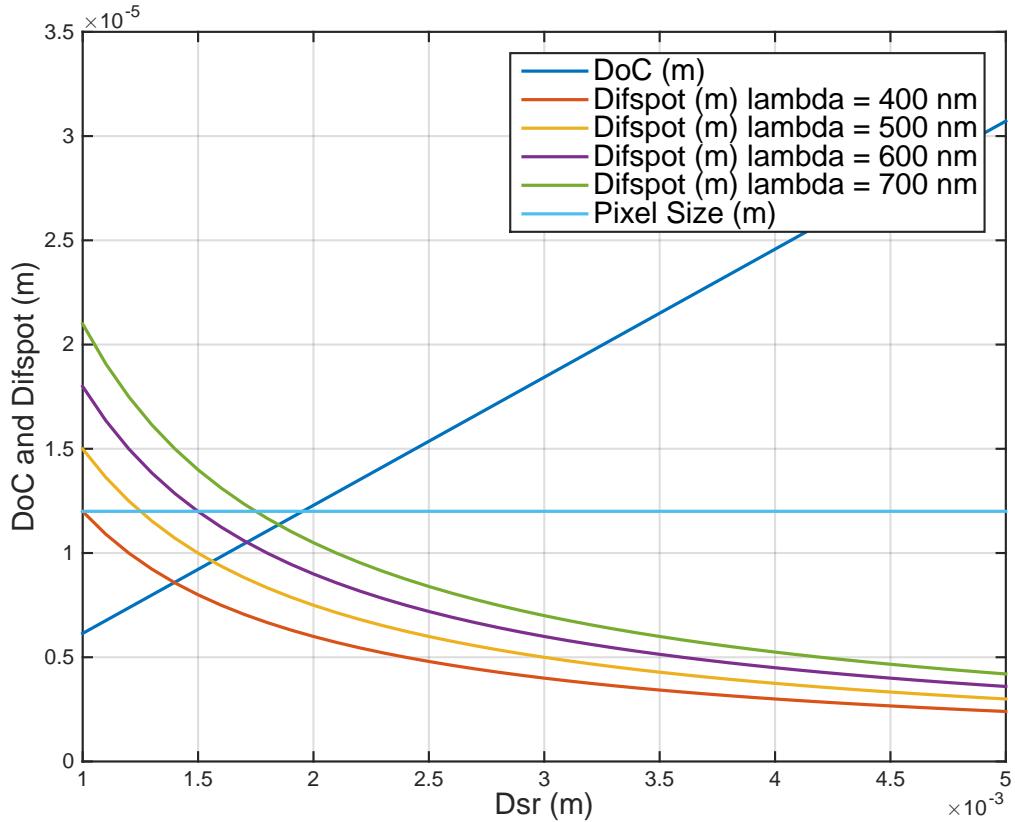


Figure 2.2: Diameter of confusion and diffraction spots as a function of the diameter

Then, the depth of field can be deduced:

$$DoF = \frac{2 \frac{f}{Dsr} DoC(m+1)}{m^2 - (\frac{DoC}{Dsr})^2} = 1.33 \text{ m} \quad \text{where } m = \frac{EFL}{rf} \text{ and is the magnification}$$

That means that the roughness of the rock analysed cannot be more than  $\frac{DoF}{2} = 66.5 \text{ cm}$  since we could not see the whole rock if not, and the 3D map will then be flawed.

## 2.1.2 Artificial Light Source

In order to carry out a 3D mapping of the target, the system needs a light source to project points which will be used by the 3D mapping algorithm. In this section, the source of light will be designed.

### 2.1.2.1 Design of the Light Source

First of all, the green has been chosen as color of the light source. Indeed, in order to detect easily the points projected on the target, a distinctive color should be used. As we can see figure 2.3, the CCD selected for the camera has the best quantum efficiency

for a wavelength around  $510\text{ nm}$ , which means that the green is the color that is the best recorded by the camera. Secondly, as the surface on Mars is mainly orange, red and brown, this color should stand out.

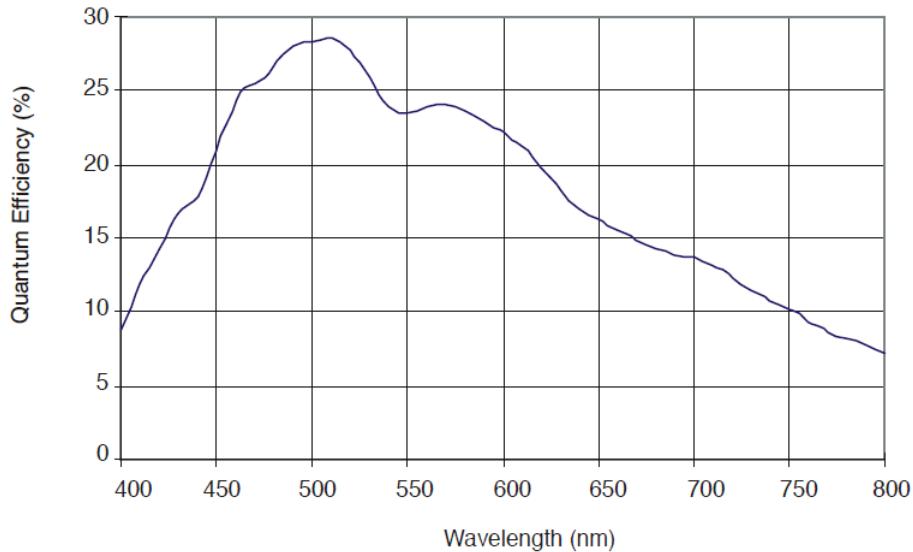


Figure 2.3: Quantum efficiency versus wavelength of the CCD

Then, as only  $300\text{ mW}$  are available to power the light source and a lot of lightning energy is needed to outshine the sunlight, a LED has been chosen. Indeed, LEDs have great energy performances. The properties of the chosen LED can be seen in Annex .2. Note that the use of a laser have been studied, but the energy cost would have been too expensive. Moreover, even if we cannot obtained a coherent light, as a laser do, with a LED, it is still feasible to get a monochromatic and diobtainrectional light. Thus, we will assume that the LED can bring enough light (proof in part on Signal Noise Ratio 2.1.2.2). Then, as the 3D mapping algorithm needs a grid of points, the LED cannot be used in its present condition. Moreover, lot of light would be lost without an optic system. Thus, a system has been designed to concentrate the luminous beams of the LED and to transform the continuous light into a grid of point. The system uses a lens and a grid (Figure 2.4).

As the LED has a diffusion angle of  $30^\circ$ (Annex .2), the lens needs to be close enough to collect all the light beams of the LED to avoid losses of energy. In order to capture as much light as possible, the diameter of the lens should follow (2.1).

$$d = f \tan \alpha \quad (2.1)$$

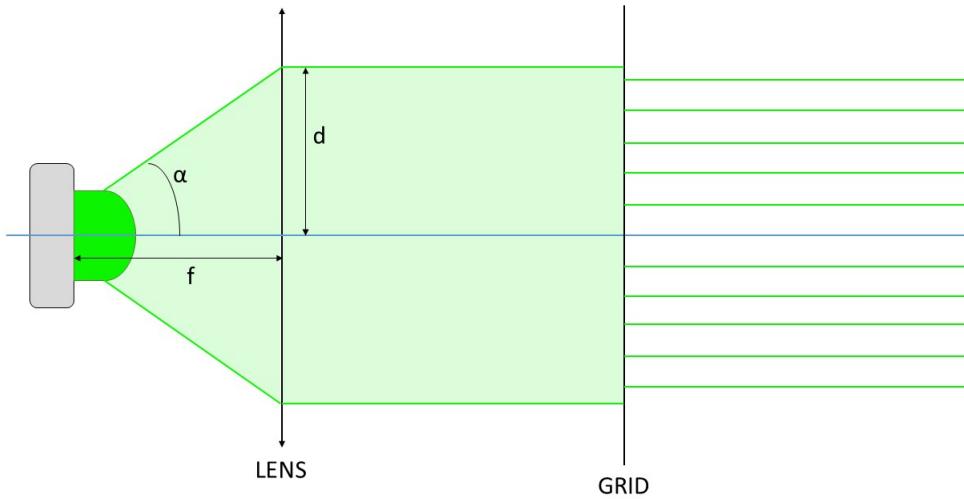


Figure 2.4: schema of the LED system

with

$$\alpha = \frac{30}{2} = 15^\circ, \text{ the half of the diffusion angle.}$$

$f$ , the focal length of the lens

Moreover, it should be pointed that there is a loss of energy because of the lens and the grid. We will assume that the energy loss coefficient of the lens *LFLed* (Loss Factor Lens Led) is 0.5 and the one of the grid *LGGLed* (Loss Factor Grid Led) is 0.3.

### 2.1.2.2 Validation of the Light Source

In order to verify whether our light source has enough power or not to outshine the sunlight, the Signal/Noise ratio (SNR) has to be calculated (see part Signal/Noise ratio 2.1.3.5). Knowing that 100 is a really great SNR, as the SNR of images recorded with this system is [32, 348], it can be concluded that the LED can be chosen as light source. Indeed, even if 32 is a low ratio, it is only when all the different poor conditions happen in the same time and it does not happen frequently. Moreover, 100 to 347 are really high ratios that permit to have the 3D algorithm working well. Also, as the artificial light source does not need to project the lighting points permanently but only during the recording of the image by the camera, that is to say during the integration time, peak powers can be used to make the point even brighter. Indeed, the artificial light source could be used in a pulse mode matched on the shutter time of the camera. Thus, the SNR could even be better. In conclusion, the design of the light source is validated.

## 2.1.3 Signal Noise Ratio

### 2.1.3.1 Irradiance

First of all, the irradiance  $F$  of the light from the sun falling at the top of the atmosphere of Mars can be calculated as following :

Conservation of energy :

$$4\pi R_{\odot}^2 F_{\odot} = 4\pi R^2 F \quad (2.2)$$

with

$$R_{\odot} = 6,956.10^8 \text{ m} : \text{solar radius}$$

$$F_{\odot} = 6,45.10^7 \text{ W.m}^{-2} : \text{energy flow of the surface of the sun}$$

$$R \in [2.06644, 2.49228].10^{11} \text{ m} : \text{distance Mars-Sun (Aphelion and Perihelion)}$$

$$F = F_{\odot} \left( \frac{R_{\odot}}{R} \right)^2 \in [502, 730] \text{ W/m}^2 \quad (2.3)$$

In this report, we will consider that the rover is working on a specific date and we will chose the one when  $R$  corresponds to the semi-major axis. In this case  $R = 2,27936.10^{11} \text{ km}$  and

$$F = 589 \text{ W/m}^2 \quad (2.4)$$

Moreover, we can assume that a part of the irradiance is absorbed by the atmosphere. Knowing that the atmosphere of Earth absorbs and scatters to space around 30% of the incident irradiance of the Sun [16], and knowing that the atmosphere of Mars is thinner than the one of the Earth (150 times less dense), we will postulate that 10% of the incident irradiance is absorbed. Thus, using (2.4) the actual irradiance  $F_a$  of the light from the sun falling on the surface of Mars is

$$F_a = \frac{90}{100} F = \frac{90 * 589}{100} = 530 \text{ W/m}^2 \quad (2.5)$$

However, this irradiance is the one of surface exposed perpendicular to the sun's beams. As Mars is a sphere, the projection need to be considered. Knowing that the weather is better into the northern hemisphere of Mars (1.1.2) and the fact that a latitude between 30 an 70 degrees is favored for a landing[6], we will assume that the rover has a latitude of 50°. This latitude corresponds to an angle of 40° between the surface of Mars and the sun's beams. Moreover, we suppose that the rover stops working when this angle

is inferior to  $10^\circ$ . Thus, the irradiance  $F_{50}$  at a latitude of  $50^\circ$  is

$$F_{50} = F_a \sin(\text{angleBeams}) \in [92, 341] \text{ W/m}^2 \quad (2.6)$$

with  $\text{angleBeams} = [10, 90 - \text{latitude}] = [10, 40]^\circ$ .

### 2.1.3.2 Target's Radiance

Considering the trajectory of the Sun into the sky of Mars and knowing that the rock target is more or less vertical to the surface of Mars, the angle  $\theta$  between the target's normal and the sun's beam is considered to be included in  $[10, 50]^\circ$ . In addition, in the optimal case (when all the optimal conditions are provided to have the maximal radiance), the BRDF of the surface of the target is assumed to be 90% Lambertian and 10% Glossy while in the worst case the BRDF will be only Lambertian. In this way, the radiance of the target  $R_T$  is

$$R_T = \begin{cases} \frac{F_{50}\alpha}{\pi} \cos \theta & \text{optimal case} \\ F_{50}\alpha\left(\frac{9}{10\pi} \cos \theta + \frac{1}{10}\right) & \text{worst case} \end{cases} \quad (2.7)$$

with

$\alpha \in [0.05, 0.45]$ , the albedo of the target 1.1.3

$\theta \in [10, 50]^\circ$ , the angle between the normal of the target and the beam of the sun

Thus,

$$R_T \in [1.44, 58.58] \text{ W/m}^2 \quad (2.8)$$

### 2.1.3.3 Target's Irradiance

Now, the irradiance of the target  $I_T$  can be calculated with

$$I_T = R_T \frac{\pi}{4} \left( \frac{Dsr}{EFL} \right)^2 \cos(\alpha_{CT})^4 \quad (2.9)$$

with

$\alpha_{CT}$  the angle between the normal of the target and the axis of the camera

$D_{sr}$  the effective lens entrance aperture

$EFL$  the focal length

However, knowing that the camera is supposed to be right in front of the target, we have

$$\alpha_{CT} = 0$$

And according to (2.9), (2.8), 2.1.1.4 and 2.1.1.3, we have

$$I_T \in [0.0271, 1.1000] \text{ W/m}^2 \quad (2.10)$$

Finally, the luminous power from the target to the camera per pixel  $W_{lum}$  is

$$W_{lum} = I_T P_s \quad (2.11)$$

with

$$P_s = 12.10^{-6} \times 12.10^{-6} \text{ m}^2, \text{ the pixel size of the CCD}$$

According to (2.10)

$$W_{lum} \in [3.900, 158.40].10^{-10} \text{ W/pixel} \quad (2.12)$$

#### 2.1.3.4 Light Source Irradiance

Now, the luminous power of the light toward the CCD from the light source reflected by the target  $W_{LED}$  needs to be determined. First, the luminous power of each point  $W_{dot}$  projected by the source is

$$W_{dot} = \frac{LFLLed.LFGLed.P_{LED}}{N_{point}} = 4,4960.10^{-4} \text{ W} \quad (2.13)$$

where

$N_{points} = 10 \times 10 = 100$  points, the number of points projected on the target

$$P_{LED} = \frac{C_{LED}\Omega}{L_e} = 0.2997 \text{ W}, \text{ the power of the LED}$$

$C_{LED} = 84 \text{ cd}$ , the power of the LED in candella (Annex .2)

$\Omega = 2\pi(1 - \cos \alpha) = 0.2141 \text{ sr}$ , the solid angle of the LED (Annex .2)

$L_e = 60$ , the luminous efficiency of the LED

Then, the irradiance of a light point  $E_{point}$  is

$$E_{point} = \frac{W_{dot}}{A_{point}} = 71.5558 \text{ W/m}^2 \quad (2.14)$$

where  $A_{point} = 6,2832.10^{-6} \text{ m}^2$ , the surface of a point.

Finally,  $W_{LED}$  can be calculated following the same steps as in the previous part 2.1.3.3 except the fact that we have to consider an angle  $\theta_{Led} = 10^\circ$  between the light source and the normal of the target. Thus, the luminous power per pixel received by the CCD from the source light is  $W_{LED} = [3.033, 33.271].10^{-12} \text{ W/pixel}$ .

### 2.1.3.5 Signal/Noise Ratio

Three different cases will be studied. In the first one, we examine the case of a target illuminated by the sun without the use of an artificial light source, the second one considers the use of an additional source by night and the last one, which is the case that we need to consider for our rover, studies the use of the artificial light source by daylight. In order to calculate the Signal/Noise ratio, the different noises need to be determined. Three will be taken into account: the readout noise, the dark current noise and the noise from the sun light.

The three different cases have the readout and the dark current noise in common which are given by the datasheet of the CCD (Appendix .4).

$$\delta_{readout} = \frac{RMS}{N_{pixel}} \quad (2.15)$$

$$= 2,3842.10^{-05} \text{ el/pixel} \quad (2.16)$$

$$\delta_{dark} = \left( 2,55.10^{15} N_{dc0} T_s (P_s.10^2)^2 T^{\frac{2}{3}} e^{-\frac{E_g}{2kT}} \right)^{\frac{1}{2}} \quad (2.17)$$

$$= 0.0067 \text{ el/pixel} \quad (2.18)$$

where

$RMS = 25 \text{ el}$ , the RMS readout noise .4

$N_{pixel} = 1024 \times 1024 \text{ pixels}$ , the number of pixel of the CCD.4

$N_{dc0} = 30 \text{ pA/cm}^2$ , the dark current level at 20°C .4

$T_s = 0.1 \text{ s}$ , the shutter time

$P_s = 12.10^{-6} \text{ m}$ , the pixel size .4

$T = 197 \text{ K}$ , the temperature

$$E_g = 1.1557 - \frac{7.10^{-4}T^2}{1108 + T}$$

$k = 8.62.10^{-5} \text{ eV/K}$ , the Boltzmann's constant

### First case

In this case, the signal that needs to be considered is the sun light reflected by the target. Therefore, the readout and the dark current noises are the only two which need to be considered and the noise is

$$N = \sqrt{\delta_{readout}^2 + \delta_{dark}^2} = 0.0067 \text{ el/pixel} \quad (2.19)$$

Then, the number of photons per shutter time  $N_p$  corresponding to the radiance of the target is

$$N_p = \frac{1}{\lambda_{max} - \lambda_{min}} \int_{\lambda_{min}}^{\lambda_{max}} \frac{W_{lum}ts}{\frac{h.c}{\lambda}} d\lambda \in [1.177, 47.810].10^6 \text{ photons/pixel} \quad (2.20)$$

where

$ts = 0.1 \text{ s}$ , shutter time

$h = 6,6263.10^{-34} \text{ J.s}$ , Planck's constant

$c = 3.10^8 \text{ m/s}$ , velocity of light

$\lambda \in [400, 800] \text{ nm}$ , wavelength of the sunlight

The number of photons per pixel going to the lens  $N_{CCD}$  is

$$N_{CCD} = \frac{\pi \left( \frac{D_{sr}}{2} \right)^2}{2\pi(r)^2} N_p \in [0.1328, 5.3936] \text{ photons/pixel} \quad (2.21)$$

where  $r = 2 \text{ m}$ , the distance between the camera and the target

The number of photons per pixel to the lens registered by the CCD  $Nen_{CCD}$  is

$$Nen_{CCD} = N_{CCD} \int_{\lambda_{min}}^{\lambda_{max}} CCDqe(\lambda).alphaLens(\lambda) d\lambda \in [0.0208, 0.8457] \text{ photons/pixel} \quad (2.22)$$

where

$CCDqe$ , the quantum efficiency of the CCD, part 2.3

$alphaLens$ , the pass band efficiency of the lens

Finally, according to (2.19) and (2.22) we obtain the Signal/Noise ratio

$$\frac{S}{N} = \frac{Nen_{CCD}}{N} = [3.1, 126.8] \quad (2.23)$$

Knowing that 100 is a really great Signal/Noise ratio, we can conclude that the system could record images of satisfactory or high quality most of the time. However, if the different poor conditions accumulate, the quality of the image will greatly decrease.

### Second case

In this case, the signal that needs to be considered is the light of the artificial light source reflected by the target. Therefore, the noise is still the same as in the first case. Proceeding in the same way as into the first case 2.1.3.5 but considering only the wavelength of the LED ( $520.10^{-9} \text{ m}$ ), the number of photons per pixel from the artificial light source going toward the CCD is  $Nen_{LED} = [2.4556, 26.9410] \text{ photon/pixel}$ .

And the Signal/Noise ratio is given by

$$\frac{S}{N} = \frac{Nen_{LED}}{N} = [368, 4040] \quad (2.24)$$

It can be deduced from this  $SNR$  that the system light source plus camera would work perfectly by night. Indeed, the minimum of this ratio is 368 which is larger than 100, an excellent ratio. Working by night would be the best scenario but the day work needs to be considered.

### Third case

In this case, the current case, it will determined if the artificial light source is powerful

enough to outshine the sunlight. Thus, the sunlight is now considered as a noise, the shot noise.

$$N = \sqrt{\delta_{readout}^2 + \delta_{dark}^2 + \delta_{shot}^2} = [0.0219, 0.8457] \text{ el/pixel} \quad \text{with } \delta_{shot} = n_{enCCD} \quad (2.25)$$

As the signal and the noise are ranged because of the different conditions possible, the SNR should be

$$SNR = \left[ \frac{S_{min}}{N_{max}}, \frac{S_{max}}{N_{min}} \right] \quad \text{with } S = N_{enLED}$$

However, it can be noticed that one parameter, the albedo  $\alpha$ , is varying for both lights, the sun and the artificial light source. Indeed, if the target has an albedo  $\alpha_{min}$ , this albedo needs to be considered in both calculation of the  $N_{enCCD}$  and  $N_{enLED}$ . Therefore, the calculation of the  $SNR$  needs to take into account this correlation and we have

$$SNR = [32, 348] \quad (2.26)$$

It can be concluded that the artificial light system is powerful enough to outshine the sunlight. Moreover, the quality of the image of the points should have an excellent quality, which would permit the 3D mapping algorithm to work perfectly. Also, even if the worst conditions are gathered (minimum albedo, sunset), the  $SNR$  is still equal to 32, which is acceptable.

#### 2.1.4 Summary and Comparison with MER Cameras

To validate the coherence of the characteristics of the designed camera, a comparative table 2.1 with the MER cameras Navcam (Navigation Camera) and Pancam (Panoramic Camera) [8] can be found below.

Table 2.1: Recap chart of the Designed Camera and Comparison with Navcam and Pancam (MER Cameras)

Features	Designed Camera	Navcam	Pancam
CCD			
Pixel size	$12 \times 12$ microns	$12 \times 12$ microns	$12 \times 12$ microns
Resolution	$1024 \times 1024$	$1024 \times 1024$	$1024 \times 1024$
Spectral range	[400 - 800] nm	[600 - 800] nm	[400 - 1100] nm
Readout noise, $55^\circ$	25 electrons	25 electrons	25 electrons
Optical properties			
Focal length	12.14 mm	14.67 mm	43 mm
Entrance pupil diameter	1.9 mm	1.25 mm	2.18 mm
FOV	$26.6^\circ \times 26.6^\circ$	$45^\circ \times 45^\circ$	$16^\circ \times 16^\circ$
Depth of field	0.67 - 2 m	0.5 m - infinity	1.5 m - infinity
Best focus	1 m	1 m	3 m

The CCD parameters are similar except for the spectral range. As Pancam mission is to investigate Mars terrain and obtain color images of any information useful to learn more about the Red Planet, that seems logical that the spectrum is wider than the one we choose. The navigation camera provides a  $360^\circ$  view of the area where is located the rover. The spectrum was reduced using filters to allow a higher spectral responsivity. For the optical features, it can be noticed that the parameters have approximately the same size. The depth of field for our camera is restricted but it is due to our aim: stabilizing the camera in front of a rock and carrying out its depth map. We can conclude that since the MER cameras accomplish well their task, the designed camera which is comparable to them should also be capable of acquiring good images on Mars.

### 2.1.5 Cost and Power Consumption

The price of a camera depends of its characteristics, especially, of the CCD sensor. The chosen one has large pixels and hence will capture more light than another CCD with half of our pixel size. That means that its performance should be better and thus, the designed camera could be quite expensive. As the exact cost of high quality camera is often not provided, and as finding a similar model to the one planned above is very difficult, it is decided to only determine a price range. Some websites such as ThorLabs have CCD cameras with a resolution of one Megapixel but with half the pixel size desired. As their cost is between 5.000 and 9.000 €, we estimate our camera at around 10.000 €. Knowing that the Nasa has planned a budget of \$1.5 billion for the new Mars Rover

Mission of 2020, this price represents only  $0.73 \cdot 10^{-3} \%$  of the total mission. Taking into consideration that the fee for the artificial source (LED and lens) is negligible compared to the camera cost, it can be concluded that the amount of money needed for buying the designed camera is reasonable.

Regarding the power consumption of our system, it is assumed that the camera could be powered with 2.15 Watts since it is sufficient for supplying the MER Cameras. On the other hand, according to its datasheet (Appendix .2), the LED consumption is of 220 mW. On the report of the NASA, a Martian rover needs around 100 Watts and they are provided by solar panels and rechargeable Lithium batteries. The system will then use around 2.4 % of the rover power. It is tolerable since the camera will not take pictures all the time. It can be argued that the processor consumption should be added. However, the other actions of the robot are also controlled by the CPU, so it is not taken into account here, since it is hard to quantify the power utilized by the algorithms to compute the depth mapping.

## 2.2 Distance Camera-Target of One Point

Before starting to establish a whole 3D mapping, a more simple case can be studied first and it will be shown, in this section, how to determine the distance to only one point. Indeed, the distance between the camera and a point of the target can be easily determined thanks to a little bit of trigonometry.

The context of this case is simple, the camera is recording images of the target while the artificial light source is projecting a point on the last one (see figure 2.5).

According to the figure 2.5,

- $z$  is the distance to determine.
- $\alpha$  is the angle between the focal axis of the camera and the focal axis of the light source.
- $a$  is the distance between the focal axis of the camera and the lighting point.
- $D$  is the distance between the camera and the lighting point when this one is on the focal axis of the camera.
- $d$  is the distance between the camera and the light source
- $p$  is the distance between the focal axis and the image of the lighting point on the CCD

Thus, we have

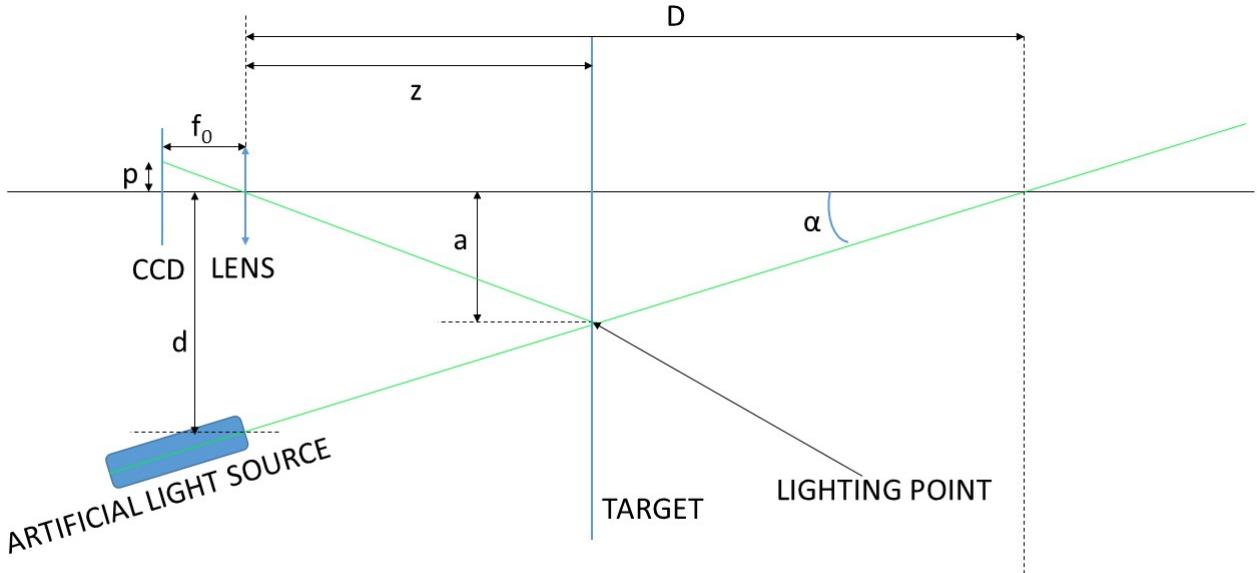


Figure 2.5: schema of the camera recording images of the target on which there is a lighting point from the artificial light source

$$d = D - \frac{a}{\tan \alpha}$$

However,  $\alpha$ ,  $a$  and  $D$  are not known yet, that is why a calibration phase is needed (see Figure 2.6). The first parameter,  $\alpha$ , is determined during the installation of the light source and the camera on the rover. Then, during the calibration, two steps can be identified. During the first one, a target is positioned such as the lighting point is on the center of the image recorded by the camera. Thus,  $D$  can be measured. During the second step, a target is positioned such as the lighting point is right on side (left or right) of the image recorded by the camera (see Figure 2.6). Therefore,  $a_0$  can be measured and we have

$$\frac{a_0}{p_0} = \frac{z_0}{f_0} \quad \text{and} \quad \frac{a}{p} = \frac{z}{f_0}$$

Thus, we have

$$a = \frac{p z a_0}{p_0 z_0}$$

And finally,

$$z = \frac{D p_0 z_0 \tan \alpha}{p_0 z_0 \tan \alpha + p a_0}$$

The distance  $p$  and  $p_0$  can be determined using the pixel size  $P_{size}$  such as

$$p = P_{pixels} P_{size}$$

but the pixel size  $P_{size}$  can be simplified and the final formula is

$$z = \frac{D p_{pixels\_0} z_0 \tan \alpha}{p_{pixels\_0} z_0 \tan \alpha + p_{pixels} a_0}$$

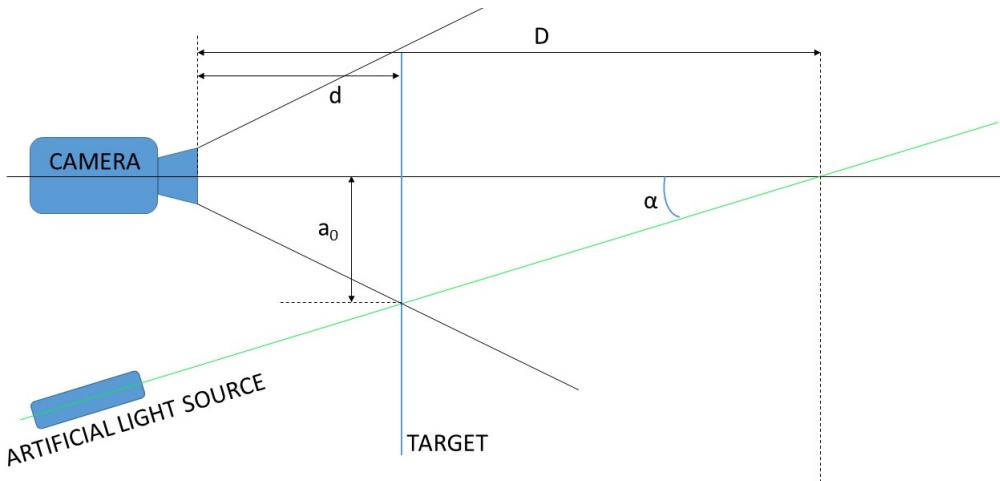


Figure 2.6: schema of the calibration

However, a calculation of the distance camera - target using a simpler calibration can be found thanks to triangulation. Indeed, using the Thales' theorem such as  $\frac{z}{f_0} = \frac{a}{p}$  (see Figure 2.5), it can be determined that the distance  $z$  camera - point on the target is

$$z = \frac{f_0 d}{p + f_0 \tan \alpha}$$

Thus, only the first step of the calibration is needed because only  $\alpha$  and  $d$  need to be determined.  $f_0$  is a characteristic of the lens and  $p$  is determined thanks to the CCD

characteristics ( $p = p_{pixels} P_{size}$ ).

$$z = \frac{d}{p_{pixels} \frac{P_{size}}{f_0} + \tan \alpha} \quad (2.27)$$

## 2.3 3D Map

In order to find the 3D map of the target, we use the triangulation method seen in the previous section on a grid of lighting points. To do so, for each lighting point, we have to use two different angles  $\alpha$  and  $\beta$ . Previously,  $\alpha$  was the angle between the focal axis of the camera and the beam but now it is the angle between the focal axis of the camera and the beam projected on the plane  $y=0$  and  $\beta$  is the angle between the focal axis of the camera and the beam projected on the plane  $x=0$  (see figure 2.7). In the same way, the distance  $d$  camera-artificial light of source becomes  $d_x$  and  $d_y$ .

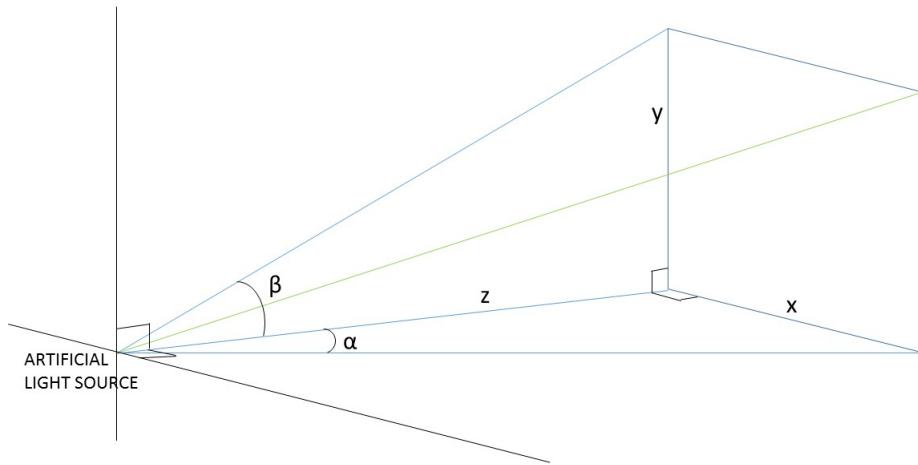


Figure 2.7: schema of a beam of the artificial light source

Thus, the new distance camera-target is

$$z = \frac{d_x}{p_{pixels,x} \frac{P_{size}}{f_0} + \tan \alpha} \quad (2.28)$$

Then, to find the (x,y) coordinates of the point, we have

$$x = z \tan \alpha - d_x \quad (2.29)$$

$$y = z \tan \beta - d_y \quad (2.30)$$

We subtract  $d_x$  and  $d_y$  in order to change from the artificial light source system of axis to the camera system of axis.

## 2.4 Image Processing

To carry out the 3D map of a Martian rock, the coordinates of the points which are projected by the LED on its surface need to be determined on the images acquired by the camera. They are obtained by detecting the points thanks to their color, by distinguishing them, and by applying the centroid algorithm to get their center. First, the color detection method will be addressed, before focusing on the differentiation and the centroiding.

### 2.4.1 Color Detection

To increase the robustness of the algorithm, the color of the light beams was chosen to have the maximum contrast with the color of the rock. Mars having a reddish color, a bright green light seemed adapted. Moreover, as it is explained in the scene analysis part, the CCD sensor is more sensitive to the wavelength corresponding to the green (around 520 nm). Thus, the color detection algorithm aims to find bright green spots in the camera video stream in real time. It is implemented in C with OpenCV library.

For unit testing, the image analysis is performed by extracting frames from the video stream obtained thanks to the computer webcam. These images are processed very quickly, which allows a real time analysis. The details of the method employed are given below. For the explanation, the figures are obtained for the detection of a blue box (the H value was changed for this purpose), but the algorithm works similarly for green light beams.

To begin with, a low-pass filter is applied to the image to handle to smooth it and to reduce the noise. Once achieved, the image is converted from RGB to HSV (see Theory Section) in order to get a better control of the colors and intensities of the image pixels. The range of the HSV parameters can be determined to isolate the bright green pixels from the rest of the image. The equivalent hue is between  $60^\circ$  and  $180^\circ$ , taking a large scope to be sure to isolate the good color component. As the luminous points will be bright, the range of the saturation and the luminosity should be chosen around the upper values. To adjust the values, several trackbars are added to the image, see figure 2.8 and the modifications are made in real time. All the values under the thresholds defined are irrelevant. Thus the pixels of the binary image which corresponds to the HSV thresholding are set to 1 (white) for the zone where the light green points are recognized and to 0 (black) for the background, the rest of the image. This image is then easier to deal with because it is only composed of two values, black and white. Erosions and dilations can be applied respectively to separate connected light beams and remove noise or to fill holes in the points. The number of erosions and dilations are selected thanks to trackbars once again

(figure 2.8) and they are computed with the OpenCV functions with a 3x3 rectangular structuring element. The size of the kernel of the low-pass filter is also controllable with the trackbars.

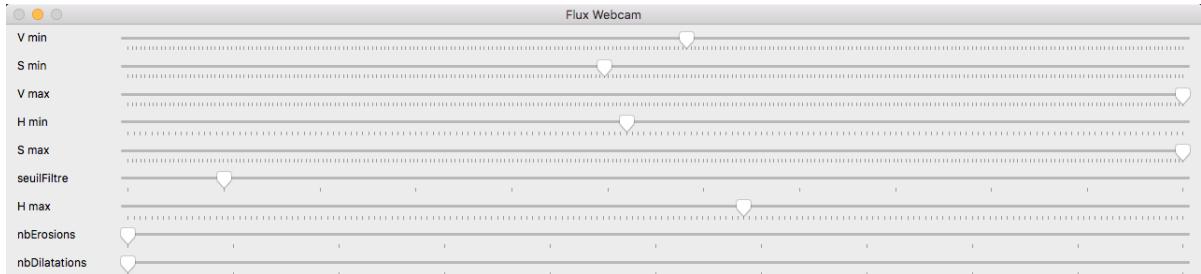


Figure 2.8: Trackbars for the threshold of the filter, the HSV parameters and the number of erosion and dilation

The calibration is carried out through the trackbars and the result is displayed on the video stream with the addition of contours to facilitate the tuning (figure 2.9). *cvFindContours* is used to search the contour of the light spots. The detection is based on the same principle that it has been seen during the exercises in the lab. The algorithm looks for the first pixel belonging to the point and then follows the contour pixel by pixel. Once the threshold of the low-pass filter, the HSV values and the number of erosion and dilation are established, the calibration is completed. For the unit testing with the blue box, we only focus on the color detection and the centroiding. Thus, no erosion nor dilation are added and the filter kernel is set to one pixel, which does nothing.

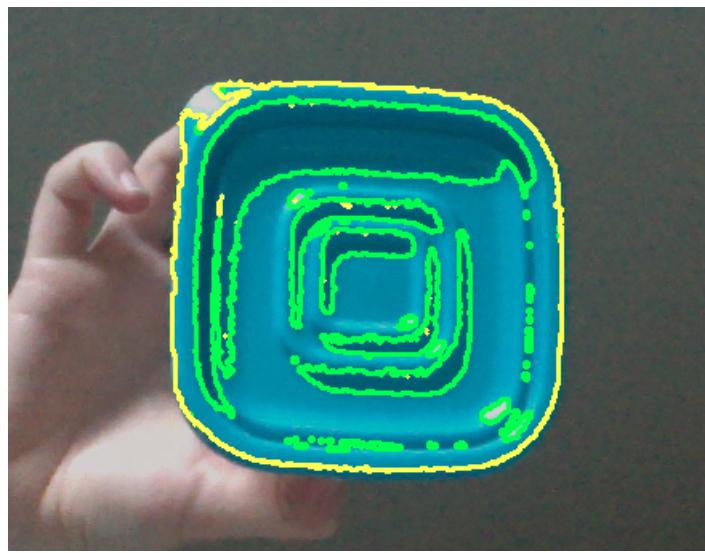


Figure 2.9: Contours of the detected object

In order to compute the centroid of the beams spots, it is needed to retrieve their intensity. To this end, the color detection should return an image with the light points in HSV color on a black background. To do so, the binary image is multiplied by the HSV

image. The multiplication is explained by the pseudo code below.

```

function MULTIPLICATION(binary image, color image)
  for each pixel in the binary image do
    if the pixel value is white then
      pixelHSVColor  $\leftarrow$  the color of the pixel in the HSV image
      the color of the pixel in the final image  $\leftarrow$  pixelHSVColor
    else
      the color of the pixel in the final image  $\leftarrow$  black color
    end if
  end for
end function

```

The final image obtained is shown figure 2.10. Each white pixel corresponding to 1 in the binary image is replaced in the final image by the HSV value of the original image and the background is set to black like in the binary image.



Figure 2.10: Object detected in HSV color system on a black background

## 2.4.2 Beams Distinction

In the previous part, a unit test was performed with a blue box to check the efficiency of the color detection algorithm. However, the aim is to project a grid of 10 x 10 green points on a rock. Once they are extracted from the rest of the image with the same technique described above, they need to be separated. Indeed, for now, we only have the coordinates

of the pixels assumed to belong to the beam spots. To look for their center of mass, the pixels which constitute each point have to be gathered.

It should be noticed that, as no pattern other than an unicolor grid is used to carry out the depth mapping, the green points cannot be distinguished from each others. Thus, this implementation does not take into account the position of each beam spot, which could be problematic if the deformation is too important. Indeed, if a point is slightly linked to another or if it has moved and is a bit above or under the others, it is still possible to retrieve its original position. It is only if the change is too important (permutation of dots for example), that it will be very complicated to get the correct location of each beam spot. To solve this problem, a solution is described in the Pattern Disorders paragraph 1.2.2.1 in the Theory Section, which would assure that no spot has been switched. Nevertheless, this method requires a pattern projection and is dismissed in a first approach.

To differentiate each point, a recursive method was implemented using C++ which consists to isolate a point using neighbouring pixels. The pseudo code explaining the technique can be found below:

```

function FINDPOINTREC(table, pixel, information to scan the image)
    ▷ table contains the coordinates of the pixel belonging to the current point
    if the pixel value is not black then
        add to table the pixel coordinates and its value
        the pixel value in the image  $\leftarrow 0$ 
        findPointRec(table, left neighbouring pixel, image information)
        findPointRec(table, right neighbouring pixel, image information)
        findPointRec(table, above neighbouring pixel, image information)
        findPointRec(table, below neighbouring pixel, image information))
    end if
end function

```

The image is scanned and it is assumed that a new point begins each time that a green pixel is detected. A table to collect the coordinates of pixels belonging to the new beam spot is created and given in parameter to the recursive method. The first pixel position is added to the table, its value is set to 0 in the image and the recursive method is repeated while neighbouring pixels are found. At the end, a 2D table is obtained where the rows correspond to the spots and the columns to each pixel belonging to the point. The centroiding algorithm can then be processed to determine the center of mass of each spot and compute their distance to the camera. We can note that, for now, the points are separated but if they were not perfectly aligned on the projection, the rows of the 2D table are not in the same order than the original projected points.

### 2.4.3 Centroiding

The 3D map can be performed with the coordinates of the light points seen by the camera. They are determined by estimating the centre of the spots, and to do so, the centres of mass, also called the centroids need to be calculated in the final image acquired by the color detection. In order to get it, a barycentre of the pixels belonging to the beam spot is carried out weighted by the intensity of the pixel (S value in HSV model).

$$x_c = \frac{\sum_x x \cdot I(x, y)}{\sum_x \sum_y I(x, y)} \quad (2.31)$$

$$y_c = \frac{\sum_y y \cdot I(x, y)}{\sum_x \sum_y I(x, y)} \quad (2.32)$$

Where  $x_c$  and  $y_c$  are the coordinates of the centre of mass of the spot,  $x$  and  $y$  the coordinates of each pixel belonging to the spot and  $I$  their corresponding intensity.

The intensity of each pixel is obtained by splitting the canals of the image into three to retrieve an image with only the Saturation values. Then the formulas (2.31) and (2.32) are computed. A white cross is displayed on the centroid in real time as it can be seen on the figure 2.10 or below figure 2.11(b) on a simple grid image 2.11(a).

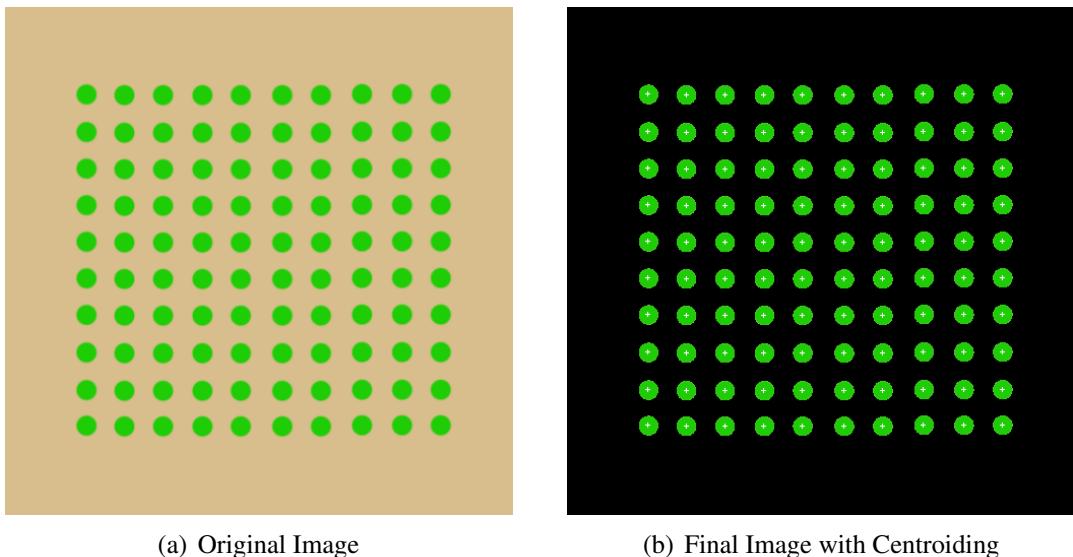


Figure 2.11: RGB images of a grid of 10 x 10 green points without any noise

In a real application, the grid will not be as perfect as previously after the projection on the rock. Small light spots could be detected as false positive, two points could be slightly linked or some spots could be displaced. In order to deal with these issues, three solutions are implemented:

- During the separation step, if a point has an insignificant number of pixel, it is dismissed. The threshold can be adjusted knowing the expected size of a point.

- Morphological operations (see Theory Section) such as openings are used to disconnect two linked points. The number of erosion and dilation to carry out to separate the points without loosing too much information about the points have to be calibrated.
- A sort algorithm is implemented to obtain the centroids of the beam spots from left to right and from the top to the bottom. It is assumed that all the points projected are detected by the color detection algorithm. Otherwise, without pattern other than an unicolor grid, it is not feasible to know which points have disappeared. As we have a 10x10 grid, the algorithm is looking for the ten centroids of the beam spots which are the most close to the top of the image. They constitute a new row of the sorted table of centroids, being classified by increasing horizontal position. These centroids are then removed from the original table and these steps are reiterated until the ten sorted rows of the grid are added to the sorted table.

These techniques are tested on the same grid as above, adding some noise (see figures 2.12(a) and 2.12(b)). To choose the adapted number of erosion and dilation to effectuate the opening, the size of the points have to be taken into account. Indeed, if too many erosions are carried out, all the beam spots will be removed. On the contrary, if not enough are accomplished, two linked spots will not be separated. Each point of the grid is composed of around 300 pixels. As we want to keep enough pixels to compute an accurate centroid, it is needed to keep at least 50 pixels. Four erosions allow to reach this result, followed by the same number of dilation to retrieve nearly the original number of pixels and to accomplish the opening. To make sure that the opening will only slightly change the centre of mass of the points, the centroiding method was performed on figure 2.11(a) with and without opening. The coordinates of the centroids were exactly the same in the two cases. The opening has preserved the information. It will be assumed that even if there is a slight centroiding error after the opening, it is insignificant.

Several types of noise are added and are denoted figure 2.12(a).

- The first one shows that small changes of green color on a point or inside the same point do not affect the color detection, but could have an impact on the centroid as it is weighted by the intensity of the color. This kind of noise may appear if the albedo of the target is not uniform for instance.
- The second noise is a spot of a green-brown color. As it is expected, it is not detected by the color detection algorithm. It may happen, as for the third and fifth noises, if the target contains green minerals for example.
- The same spot (3 on the figure) but with a color close to the grid points is added to the image. It is detected as a point but its size is under the threshold chosen and it

is dismissed as it can be seen on the figure 2.12(b). The opening also ensures that the unexpected spot disappears.

- The fourth noise represents two linked points. Thanks to the opening, which reduces the size of the beam spots before increasing them but not as much as they were originally, the connected points are well separated on the final image. This noise can happen if the relief of the target is abrupt.
- In 5, a smaller green spot (which is noise) is linked to a real point. The spot is tiny and even if it is detected by the color algorithm, the opening is efficient enough to delete it as it is on the edge of a bigger beam spot.
- Finally, the last kind of perturbation tests if the sort algorithm is effective. Indeed, the point (6) is shifted upward and as the program looks for the green spots row by row, its centroid should be in first position in the second row of the table. Thanks to the sort algorithm, we can notice (by checking the table) that it is not the case and that it is displaced to the right box. This kind slide of the dots will happen each time because the target will not be plane.

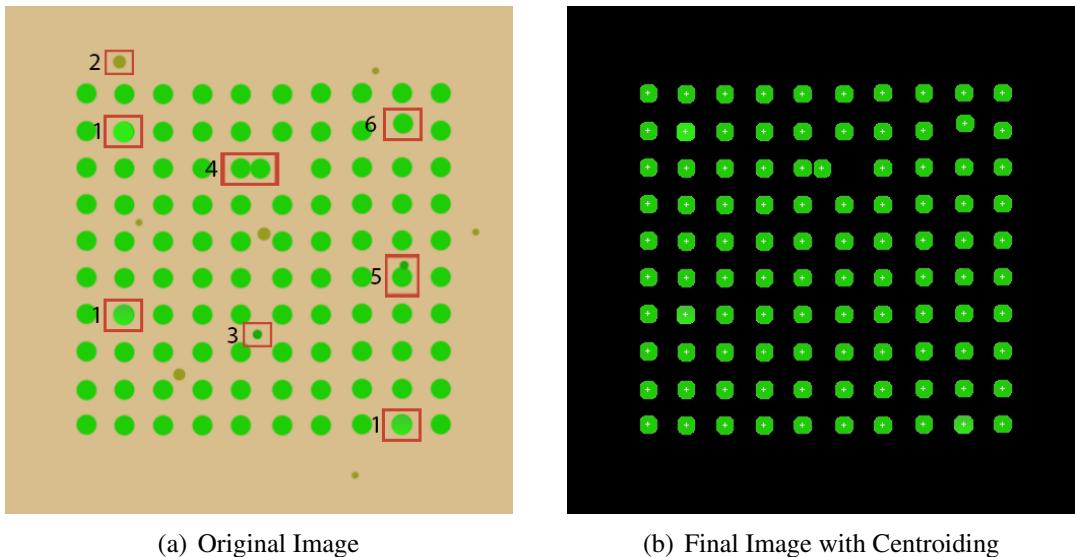


Figure 2.12: RGB images of a grid of 10 x 10 green points with noise

It can be concluded that the algorithm is robust enough to handle some noise on computed images. Nevertheless, the deformation and the unexpected light spots or reflets on the rocks could be worse than expected and the algorithm may be incapable of finding the centroid of each point, with no false positive or false negative. To ensure the robustness of the algorithm, integration tests with a real system have to be carried out.

# Part 3

## Experimental Results

To validate the system and algorithms previously defined, several integration tests are carried out. The camera used and the additional source of light are not the same as the one described above, and they are performed on Earth, thus the verification of the system will not be complete. Nevertheless, it will be enough to, at least, confirm that the approach is achievable and needs to be further investigated.

### 3.1 Equipment Employed for the Experiments

To carry out the experiments, we should build the system designed in the previous part. However, to saving time and money, it is decided to use an alternative equipment, easier to obtain, with a behavior similar to the chosen one. The components of the new system are described below.

#### 3.1.1 Lab Webcam Philips SPZ2000

First, to carry out all the experiments, a camera to acquire the images of the scene is needed. As the camera designed in the Scene Analysis have to be built with expensive constituents, and we didn't have a lot of time, it was decided to use the webcam in the lab as a substitute. They are not accurate and their image is blurred when the object to focus is too far from it but if the algorithms compute the right distances with such a camera, a more advanced one will be capable to obtain the same or even more precise results. However, some information such as the focal length and the pixel size of the CCD are unknown. A preliminary calibration to determine the focal length and the lens distortion will be achieved before performing the experiment.

### 3.1.2 Laser Pointer

To simulate the additional source of light, a laser pointer is used. Only one beam is needed to perform the experiment, thus no other system will be added to split the ray. The wavelength of the laser is 532 nm, like the color chosen in the design part. Its maximum output power is 5 mW which corresponds to the order of magnitude selected before. It may not be enough to outshine the sun, but it is adequate for checking the algorithms indoor. It is powered with batteries which will not be adapted on Mars as the rover should be autonomous.

### 3.1.3 Computer

On the rover, it is a micro-controller which will provide the computational power to carry out the image analysis and the calculation of the distance of the beam spots detected. As we are only testing the algorithms, and not implementing the whole system designed, a computer connected to the webcam to retrieve the images acquired is used.

## 3.2 Parameters of the Webcam

As the datasheet of the webcam employed to carry out the experiments is not exhaustive, we need to establish its focal length and the impact of the lens distortion.

### 3.2.1 Determination of the Focal Length

Two techniques are used to find the focal length: the triangulation method and the Matlab Camera Calibration Toolbox [9].

#### 3.2.1.1 Triangulation Method

The first approach consists in placing an object whose size is known at a certain distance of the webcam and in taking a picture of it. The figure 2.5 can be taken as a reference, where  $z$  is the distance camera-object,  $a$  the size of the object,  $p$  the size of the object in the image plane and  $f_0$  the focal length that we want to determine. With the Thales' Theorem, it is deduced:

$$\frac{f_0}{z} = \frac{p}{a} \quad (3.1)$$

As we measure the size of the object in the image plane in pixels, it is needed to convert it in meters. Thus, the size of each pixel of the CCD has to be known, which is not the case since the datasheet does not provide this information. However, by looking at the formula (2.27), it can be noticed that we need the ratio  $\frac{\text{PixelSize}}{f_0}$  and not  $f_0$  alone. This means that

by deriving (3.1) using  $p = P_{pixel} \times Pixel\ Size$  on the x direction, we obtain our desired ratio:

$$\frac{Pixel\ Size}{f_0} = \frac{a}{Z \times P_{pixel}}$$

The ratio is calculated for an object of 1 m, at a distance of 5.21 m, which corresponds to 147 pixels in the image plane. It is equal to  $0.0013\ pixel^{-1}$ . This ratio is used to compute the distance object-target in the experiments but as it depends a lot of parameters measured by hand, another solution, more accurate as it is based on the cross correlation of several images is considered.

### 3.2.1.2 Camera Calibration

Camera features, which include the focal length, can be obtained thanks to the Matlab Camera Calibration Toolbox [9]. Several pictures of a checkerboard are taken, changing the orientation of the checkerboard and its position in the image. The Matlab Single Camera Application App analyses these images and detects the intersections between four squares of the checkerboard as it can be seen in green on the figure 10 (Appendix .3). The origin is in yellow. Once this step achieved and the real size of one square known (39 mm here), the camera intrinsics, extrinsics and lens distortion parameters are estimated by a technique called geometric camera calibration. It uses the intersections coordinates of the checkerboard in different configurations and the factor scale to model the camera by a camera matrix which permits to transform the world coordinates into pixel coordinates on the image plan.

#### Extrinsics Parameters

The extrinsics parameters composed of a rotation matrix and a translation convert the world coordinates into the camera coordinates. As we are already doing this computation with our own method, these parameters are not relevant for us. Nevertheless, it can be noticed (see figure 11 in Appendix .3) that the Matlab functions locate accurately the position of the checkerboard. The Z coordinate is found to be around 1 m 40 and that is at that distance that the experiment was performed. Their angle seen by the camera is also precise. The correlation between several images is proved to be efficient to find the distance of a geometric image but it is not feasible on Mars soil.

#### Intrinsics Parameters

The intrinsics parameters are composed of the focal length and of the coordinates of the optical center in the image plan (see Table 3.1). The focal length is given in pixels in the X and Y directions.

Table 3.1: Intrinsic Parameters with Estimation Errors

Intrinsic Parameters	Focal Length (in pixels)	Optical Center (in pixels)
Direction x	$826 \pm 33$	$378 \pm 29$
Direction y	$805 \pm 31$	$175 \pm 34$

As  $f_x = \frac{f_0}{Pixel\ Size_x}$ , we can compare the ratio previously calculated thanks to measurements to the  $f_x$  found by calibrating the camera with the Matlab Toolbox. Taking into account the errors:

$$Ratio = \frac{1}{f_x} \in [0.00116 \quad 0.00126] \text{ pixel}^{-1}$$

Taking the mean of this fraction, the error between the two ratios is of 7 %. As even the ratio with the Matlab Toolbox is not accurate since the lower bound is increased by 8.7 % to get the upper bound, it is decided to use this calibration ratio for 1 beam spot integration testing only. Regarding the experiment with a row of points, as it was carried out for the same distance as the focal length computed by the triangulation method, the first ratio is kept.

### 3.2.2 Discussion About the Lens Distortion

The smaller the lens is, the more is the chance that lens distortion occurs. The Toolbox allows us to consider the radial and tangential distortions. According to the documentation [9], the latter "occurs when the lens and the image plane are not parallel" and the first one "when light rays bend more near the edges of a lens than they do at its optical center". To account for the lens distortion in our model, which could permit to obtain more accurate results as the webcam could not be perfect, the two distortions are applied to the distance in pixel between the center of the image and the optical center (computed by Matlab, see Table 3.1) and between the centroid studied and the optical center. The difference between the two distances distorted gives us the distorted distance between the center of the image and the centroid, used to calculate the depth map. The formula to find the distortion in the x direction can be found below:

$$x_{distorted} = x(1 + k_1 \cdot r^2 + k_2 \cdot r^4) + 2p_1 \cdot x \cdot y + p_2(r^2 + 2x^2) \quad (3.2)$$

Where x and y are the undistorted distance to the optical center, normalized by the focal length in pixel,  $r^2 = x^2 + y^2$ ,  $k_1$  and  $k_2$  the radial distortion coefficients and  $p_1$  and  $p_2$ , the tangential ones given by the Calibration App.

### 3.3 Determination of the Depth of One Laser Beam

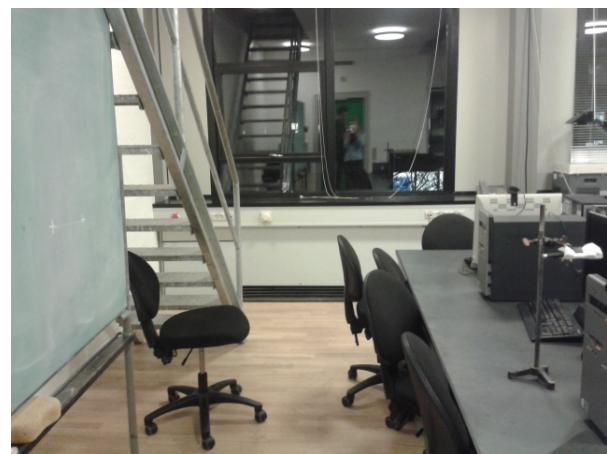
In this first integration test, we would like to establish the distance from the camera to one specific point of a surface, before checking the algorithms for several points. The general procedure is detailed below.

#### 3.3.1 General Procedure

As the angle between the focal axis of the camera and the focal axis of the light source is easier to obtain when the laser and the camera are aligned, they need to be located on the same axis for the experiment (see figure 3.1(a)). Once they are placed, the distance  $d$  between them is measured. Then a surface is required to project the laser beam. As it is more convenient to have a mobile surface to be able to move easily the target, we choose to use a blackboard for the test (figure 3.1(b)). The green is dark enough to contrast with the light green spot and the albedo of this surface is not very important which illustrates one of the worst case of the reflective power of a martian rock. The distance from the blackboard to the camera is measured to calibrate the system.



(a) System Camera-Laser



(b) Blackboard

Figure 3.1: System Camera-Laser-Blackboard

##### 3.3.1.1 Calibration

Two calibrations need to be accomplished before carrying out the 3D map of the point: setting the laser position and tuning the parameters thanks to the trackbars to detect the point.

#### Material

The webcam is connected to the computer and the calibration program is run. It only consists in acquiring frames from the webcam and displaying them on the screen of the

computer. A cross is added in the center of each image to facilitate the calibration. The purpose of this step is to obtain the spot projected by the laser at the center of the image. Indeed, once this achieved, we get easily the angle  $\alpha$  between the focal axis of the camera and the one of the light source, needed to compute the depth map. It can be deduced from the figure 2.5 that  $\tan(\alpha) = \frac{d}{z}$ , where  $d$  is the distance camera-laser and  $z$  the distance camera-blackboard.

### Color Detection

Once the laser position is set, the color detection program is started. The parameters need to be tuned to detect the bright green color of the beam spot. As we are dealing with only one point, it is not imperative to add erosion or dilation to make an opening. The noise can be discarded with only the expected pixel size of the spot in the image plane. Thus, only the HSV values are changed to match the green of the point.

#### 3.3.1.2 Distance Computation

The calibration accomplished, the distance can be calculated either in real time since frames for the webcam are acquired quickly and the processing is short, either later on, using pictures saved after the calibration steps at different positions of the blackboard.

### 3.3.2 Experiment and Results

The general procedure described is implemented with the distance camera-laser equals to 30 cm, the distance camera-blackboard equals to 137 cm, hence an angle  $\alpha$  of 12.75 °. The experiment was carried out in real time and repeated a second time, handling only pictures and taking into account the lens distortion. This is the latter result which are given below in the Table 3.2. They correspond to the image processing displayed on figures 3.2(a) and 3.2(c).

Table 3.2: Results of the depth mapping for 1 point with and without lens distortion

Distance (cm)		
Camera-Blackboard	Computed without distortion	Computed with distortion
137	136.2463	136.2557
124	125.2243	125.2153
111.5	113.2045	113.1927
98.5	100.7686	106.4445

Finally, the errors with and without lens distortion are detailed in the chart 3.3. It can be noticed that the more the blackboard is brought closer to the camera and is moved away

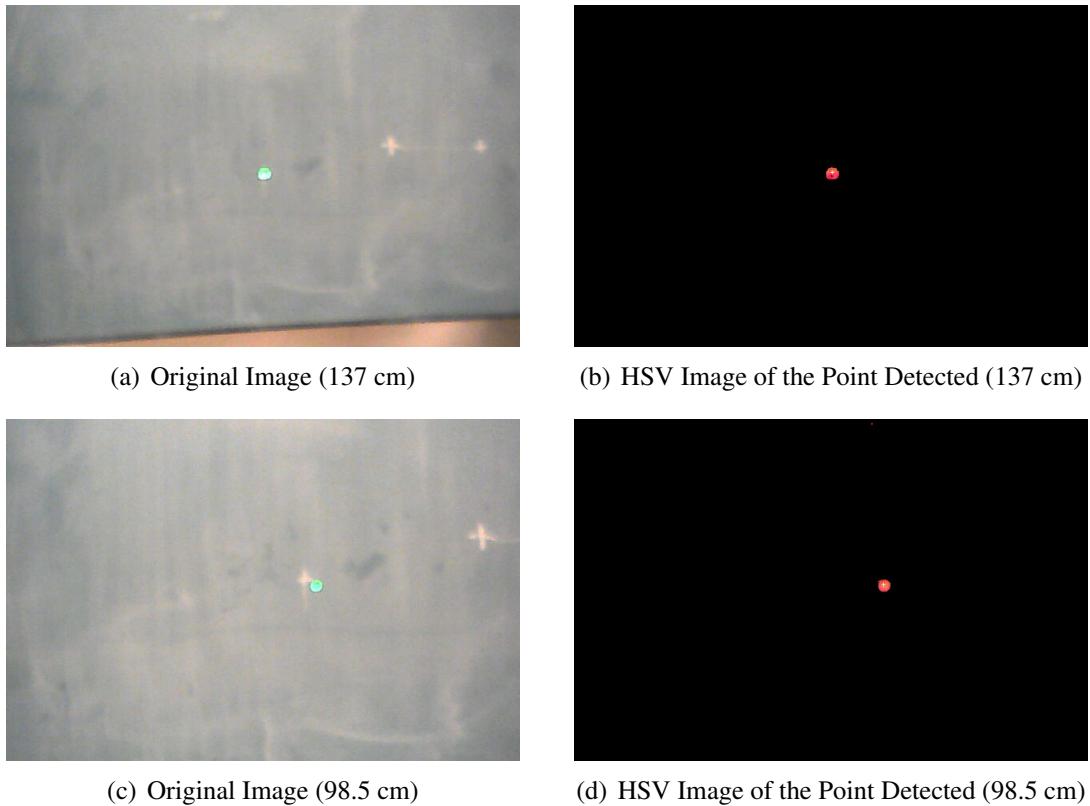


Figure 3.2: Experiment Images Acquired by the Camera and Processed for 137 cm and 98.5 cm

from its calibration position, the more the error increases. Indeed, when the blackboard is closer to the camera, the displacement of the point from the center of the image is larger (see figures 3.2(b) and 3.2(d)) and the error gets bigger with the distance centroid-center of the image. Without the distortion, the inaccuracy is between 7.5 mm for 137 cm and 22.7 mm for 98.5 cm. For the latter, it is quite a large error since the rock could be nearly flat and in order to stabilize the arm, a precise 3D map needs to be carried out. If the rock relief is in the range of the millimeter, it is likely that all the results of the distances will not do any difference between several points and the stabilization could not be achieved since no specific spot could be tracked in two images acquired at different times. The errors could be caused firstly by the fact that the experimental conditions were not ideal. The distances were measured thanks to a measuring tape but the precision was only around the millimeters, and as it was not perfectly straight between the camera and the blackboard, measurement errors could occur. Moreover, the blackboard was moved several times and could have not been perfectly perpendicular to the camera optical axis. Then, the camera images were blurred and it was not designed for this kind of processing, which biased the centroiding. Also, when the point is detected, it could have less or more pixels than the original point, used for the calibration. That means that the center of mass could be displaced and create important errors. An analysis of this error will be carried out further

in the report.

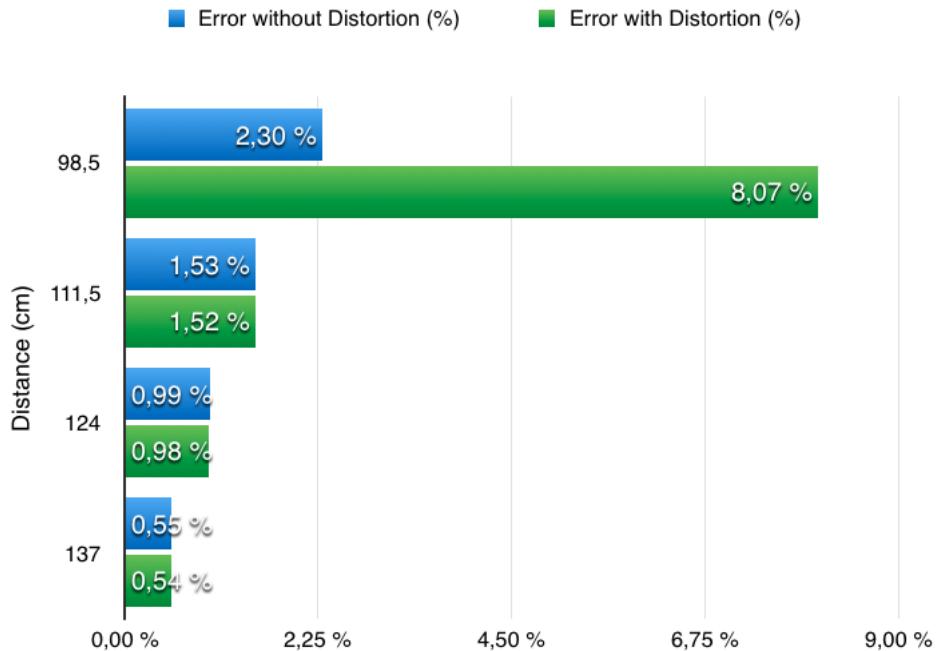


Figure 3.3: Distance Errors with and without Lens Distortion

Finally, the lens distortion have to be taken into account for explaining the results. Indeed, it can be observed that the error between the real distance and the computed one comes from the fact that the lens have distortions and that the distance measured between the centroid and the center of the image is inexact. To deal with this issue, the radial and tangential distortions parameters calculated by the Matlab Calibration Toolbox were added to the algorithm in order to compute the distance considering the case with the lens distortion. The results are also detailed in the Table 3.2 and the chart 3.3. As expected, when the distance is computed with the distance centroid-center of the image distorted, the measurement error is slightly smaller than in the case without distortion and it can be deduced that the lens distortion does not affect a lot the depth mapping. However, for the last distance, the error with distortion is not at all expected and shows either the limit of the lens distortion or that the distortion was wrongly implemented. Indeed, Matlab has estimated it for 137 cm. At this point, there were already estimation errors. As we get closer, the distorted distance increases and the errors are added. We should try to calibrate the camera at this distance or even closer and check the distortion parameters. Nevertheless, even if the error is accumulated, we should not obtain such a difference for the last distance tested. The implementation should be revised. An other solution will be to carry out this integration test with the designed camera, which should have a negligible lens distortion and be more accurate. The results should improve and the error may be tolerable.

Even if the results are not accurate and insufficient, it is a good beginning of distance

approximations and with a better material and experimental conditions, they could be improved. It should also be noticed that the time of computation is low and sufficient for real time applications. Indeed, without taking into account the lens distortion, the distance is calculated in around 35 ms and when the distorted distance is also computed, the algorithms are 1 ms slower. It is now desired to find the depth of multiple points at the same time.

## 3.4 Distances with Several Points

After the experiment with only one lighting dot, the next step is using several lighting points. Thus, the goal of this experiment is using multiple points in order to detect several distances in front of the camera.

### 3.4.1 Choices of Implementation

First of all, a grid of dots is supposed to be projected on the target. However, instead of a grid, a line of 10 dots is used during this experiment. Indeed, as the triangulation theory between a grid and a line of dots is the same and the indexing of the dots of a grid is much more difficult than the one of the dots of a line, it has been chosen to use a line in order to focus on the improvement of the robustness and accuracy of the algorithm.

Then, as the purpose of this project is the vision analysis, the focus is on the structured light algorithm and the artificial light source has not been built. Thus, in order to be able to project a line of several dots on a target, it has been chosen to use a overhead projector. Indeed, even if a laser was available (see the first experiment), after trying to diffract its beam (using a CD for instance), it appeared that the brightness of each dot was very imbalanced (central dot very luminous compared to the others) and would lead to a modification of the parameters of the dot detection. Moreover, as the artificial light source is supposed to project a grid of identical dots, the overhead projector has been chosen to respect this equality of brightness.

### 3.4.2 Progress of the Experiment

A picture of a line of 10 green dots (a green close to the one of the laser) with a reddish background (in order to represent the color of the rocks of Mars) is projected on a board using the overhead projector. Then an object is placed between the board and the camera and the several distances  $z_i$  are computed (see Figure 3.4).

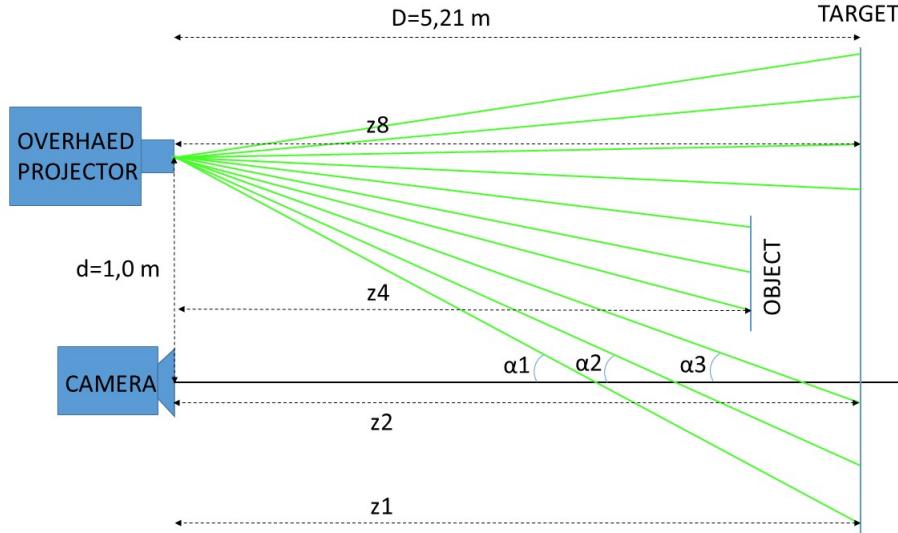


Figure 3.4: schema of the experiment

Once the script is run, the different steps are the following :

### Step 1 : Calibration

In order to find the different angles  $\alpha_i$  between the focal axis of the camera and the beams of the projector, a calibration function is called before the real time loop. This function detects the lighting dots, their centroid and use the following formula to save the different  $\tan \alpha_i$  into an array:

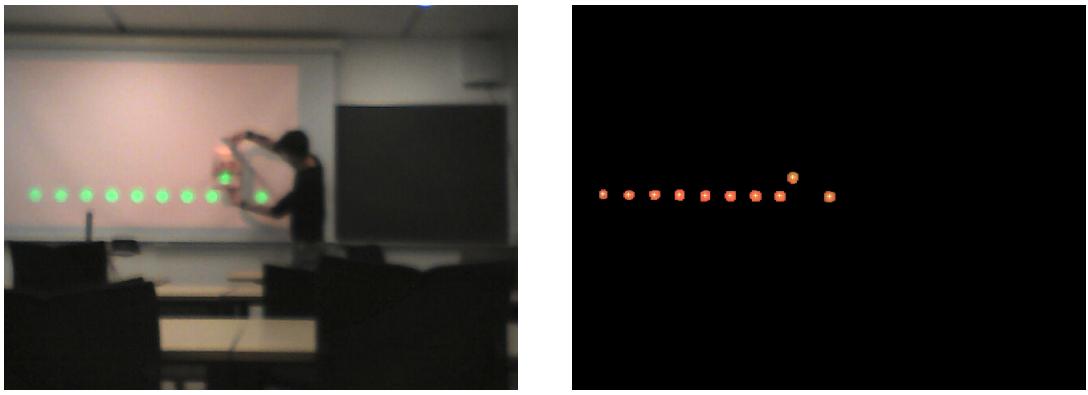
$$\tan \alpha_i = \frac{d}{D} - P_{pixels,i} \frac{P_{size}}{f_0}$$

### Step 2 : Positioning of the Object

During this step, while the script is still running, the object is placed between the camera and the board (see Figure 3.5(a)). On the one hand, as the projector is higher than the camera, it makes the 9<sup>th</sup> dot slides vertically. However, it does not affect the result as only the horizontal slide is taken into account and the *sort* function permits to keep the dots in the right order even if they slide vertically. On the other hand, a horizontal slide occurs as expected.

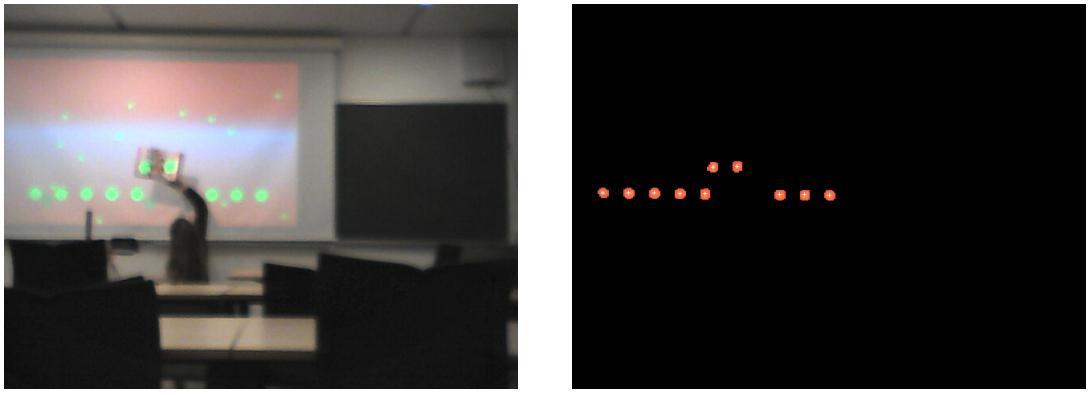
The experiment has been carried out with several distances and placement of the object in front of various dots. Moreover, some noise has been added. As we can see figure 3.6(a), the background is nuanced, smaller dots with the same green are added and dots of the same size but with others greens are added too. We can see figure 3.6(b) that the algorithm manages to filter the different noises and still detects the 10 dots.

This experiment was carried out in real time, that is to say the several distances were displayed continuously. However, some pictures were saved in order to permit people



(a) Positioning of the object in front of the 9<sup>th</sup> dot (b) Detection of the centroids

Figure 3.5: Distance object-board = 50 cm



(a) Positioning of the object in front of the 6<sup>th</sup> and 7<sup>th</sup> dots (b) Detection of the centroids

Figure 3.6: Distance object-board = 90 cm

to run the algorithm used during the experiment. Indeed, you can find into the folder '?????????' the C++ script used and pictures of objects at several distances taken by the camera. Moreover, the Power Point employed to project the dots and backgrounds are also present in order to permit the performing of the experiment in real time but in this case, the distances camera-projector  $d$  and camera-target  $D$  need to be adapted and the calibration done.

### 3.4.3 Results

The results of the experiment for the different distances object-board are presented Table 3.3 and Appendix .1 Tables 4, 5 and 6.

Table 3.3: Object in front of dots 1 and 2 with distance object-board = 0.50 m (without noise)

i	$z_i$ [m]	$z_i$ error <sup>1</sup> [%]	distance board-Object [m]	distance board-object error <sup>1</sup> [%]
1	4.69081	0.41	0.519	3.84
2	4.74072	0.65	0.469	6.14
3	5.20312	0.13	0.007	
4	5.17385	0.69	0.036	
5	5.20999	0.00	0.000	
6	5.2103	0.01	-0.000	
7	5.21342	0.07	-0.003	
8	5.22243	0.24	-0.012	
9	5.24131	0.60	-0.031	
10	5.30847	1.89	-0.098	

### 3.4.4 Interpretations

As we can see on the Table 6, the errors of the results of the experiment with noise are in the same order of magnitude than the ones without noise. It means that the algorithm is robust enough to handle the noise. Indeed, the opening permitting the selection of only the dots of the projected pattern works well. Moreover, as we can see figure 3.5(a) and 3.6(a), the images recorded are very blurred because of the camera used (webcam Philips SPZ2000). As the results are acceptable, it means that the algorithm deals also with the blur, that is to say that it could even work if the target gets a bit out of the depth of field.

However, according to the four Tables, we can see that the errors of the distance object-board is between 0.04 and 9.06 %, which can be a significant error. Indeed, an error of 9.06 % represents an experimental distance of 455 mm instead of 500 which means an error of 45 mm. For instance, our target is supposed to be a rock at a distance between one and two meters. Let us assume that one of the projected dots is on a part of the rock which is 1500 mm far from the camera. An error of 9.06 % would lead to a measured distance of 1364 mm, that is to say an error of 136 mm. To have a negligible error, the rock should have a shape such as the depth would vary over a range of at least one meter. Therefore, it would definitely not be acceptable.

These errors come from the calibration and the detection of the centroids. Regarding the calibration, we have to notice that during this experiment, it was not rigorous. Indeed,

$$^{1}error = \frac{distance_{theory} - distance_{experiment}}{distance_{theory}} 100$$

the distances board-object,  $D$  and  $d$  were measured with a measure tape which is not precise enough. Also, the focal axis of the camera was not exactly perpendicular to the target as it is supposed to be according to the delimitation of the scene. Therefore, the errors of the measurements of the distances should decrease with a precise calibration done with adapted tools. As the overhead projector was fixed on the ceiling and the focal axis of the camera had to stay perpendicular to the target, we were not able to make the distance  $d$  vary. It could have been interesting to make the angles  $\alpha$  vary in order to study the precision of the results according to these angles. Indeed, a bigger angle  $\alpha$  would lead to a bigger distance  $p$  and thus variations of the centroids would have been more negligible.

Moreover, several aspects of the detection of the centroids could be improved. The main one is that, as the brightness of the lighting dots is not uniform, that is to say it decreases with the distance from the center, the algorithm of color detection does not detect the same pixels over time, some bordering pixels are detected or not over time. This weakness of the color detection changes the position of the centroids and leads to a bigger error. Indeed, the experiment from Table 4 has been done again (using the capture of the image recorded by the camera during the experiment in order to compute exactly the same distances and get the same results) but subtracting one pixel to the distance  $p$  between the image of the object on the CCD sensor and the geometrical center of the CCD. This subtraction simulates an error of one pixel to the left of the position of centroid computed. The result of this experiment is that the new distance board-object is now 0.484 m instead of 0.455 m and the error decreases significantly from 9.06% to 3.22%. In this case, one pixel represents an error of 29 mm, that is to say an error of almost 6%. The same experiment has been done with the image of the experiment from Table 6 and the errors change from 3.51% to 0.28% and 0.04% to 3.71%. One pixel represents an error of approximately 3.77%. Finally, using the experiment Table 5, one pixel represents an error of almost 2.69%. The error caused by one pixel on the distance target-board has been calculated for 3 different values of  $\tan \alpha$  (see Figure 3.7). As we can see, when  $\tan \alpha$  is equal to 0.1919, the errors match with the values obtained during the experiment. Moreover, this graphic shows that even if  $\tan \alpha$  does not matter for big distances board-target with the target close to the camera, the error caused by one pixel is very important when the target is near the calibration distance, that is to say 5.21m. It makes sense because at such a distance,  $p$  is supposed to be zero so the error will be much more accentuated. We can conclude that, as the target is a rock with a relief inferior than 50 cm, the system needs to be accurate for short relief and the error will be minimized using a distance  $d$  camera-Artificial Light Source as big as possible.

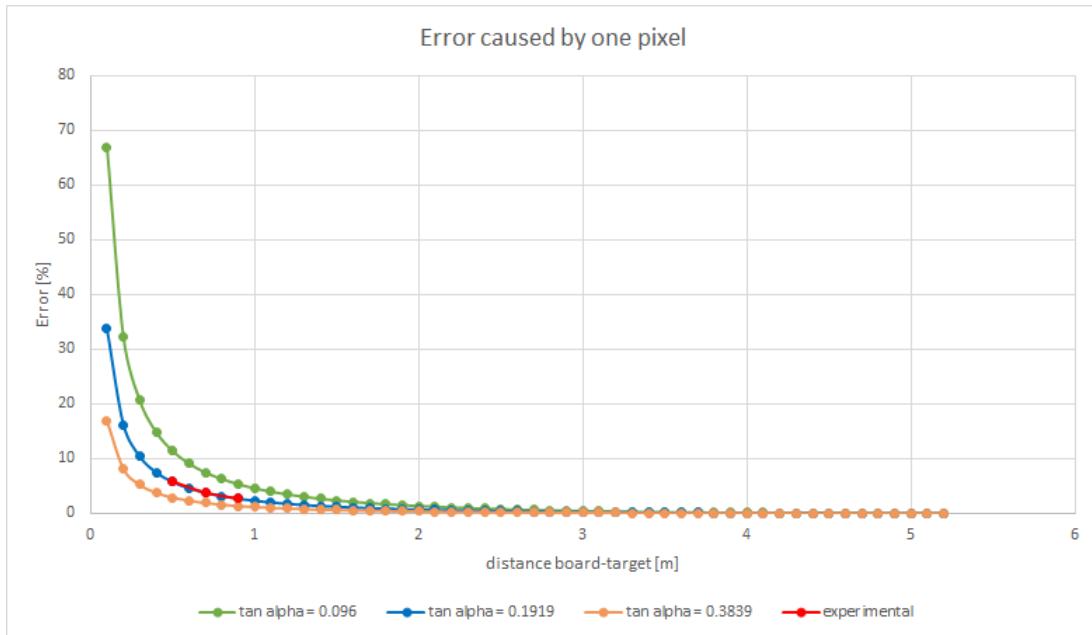


Figure 3.7: error on the distance board-target caused by one pixel

Finally, the real time computation is respected since the program needs 38 ms to find the distances of each light spots of the row.

### 3.4.5 Limits of the Experiment

The first limit of this experiment is that some pattern disorders are not taken into account. The permutation of dots that can happen with an object too far from the board has been prevented with a maximum distance board-object = 90 cm. With a greater distance than one meter, a permutation appears between the dots on the object and the ones at the left of the object but this permutation is not managed, not corrected by the *sort* function. Another disorder not tested is the merge of two dots. As the projector was on the ceiling, projecting slightly downward, the dots shifted horizontally but also vertically and as a consequence, the dots on the object were too high to merge with the dots on the board. Even if this kind of disorder was tested during the integration tests 2.4.3, it would have been interesting to test it during the experiment. A second limit is that, as this experiment does not include several aspects of the system designed (projector instead of the artificial light source, line of dots instead of a grid, ...), it cannot be considered as a test of this last one. Nevertheless, it permits to test the accuracy and the robustness of the algorithm tested. It is only an integration test.

## 3.5 Outdoor Integration Testing

The two first experiments were carried out indoor. Thus, the sunlight was not taken into account, which is quite problematic since it could make impossible the color detection as

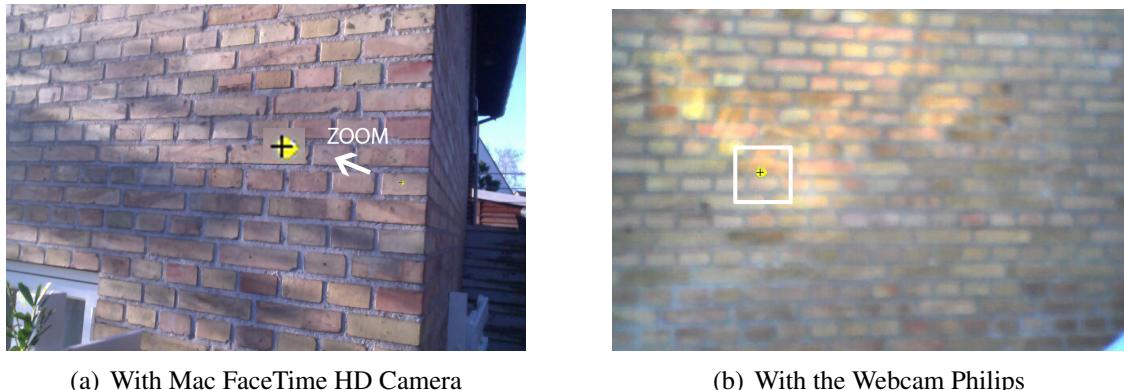


Figure 3.8: Images Acquired by the Two Different Cameras Outdoor With Sunlight

the green beam will be as bright as the sun rays on the surface. In order to check if despite the sunlight, the color detection algorithm would be capable of detecting the laser or LED beam spot on Mars, it is experimented on Earth since the sun has similar features. As the webcam used during the experiment blurs the image, another picture is acquired to make sure that it works in the two cases. The images of the test, taken around 11 am during a sunny day, are shown on figures 3.8(a) and 3.8(b).

The experiment is quite conclusive. The HSV parameters were tuned and it was difficult since the spots are small and could be mistaken as noise. However, it was feasible, as it can be noticed on the figures, where the centroid is displayed in black. It is on the point even if in the first image, it is not in the center of mass because the size of the point was reduced by decreasing the noise. Since the Signal Noise Ratio on Earth with the Sunlight as noise should be similar to the one on Mars, it can be deduced that the detection of the artificial light beam spots is achievable.

# Conclusion

We designed a system permitting the 3D analysis of a Martian rock using structured light and triangulation theory, and started the implementation by carrying out a color detection of the green dots and a calibration to compute in real time the distances camera-target.

Regarding the camera features, the system allows to work on a target between one and two meters from the rover as expected. Moreover, the depth of field is also satisfied as it is 66.5 cm and the target has a relief of 50 cm maximum. The camera characteristics could even be changed in order to reduce a bit the depth of field and increase other properties, for instance the distance camera-target. Indeed, even if this condition is fulfilled, a bigger distance would permit an easier positioning of the rover. Although the camera has not been tested, the comparison of its properties with the Navcam and Pancam suggests that it is well designed.

Considering the artificial light source, it has not been checked either but the high Signal/Noise ratio indicates that the lighting points of the grid projected on the target will be bright enough to outshine the natural light and permit to have good images and a well working structured light algorithm.

The robustness of the color detection was evaluated through different indoor and outdoor experiments. The algorithm is found to be efficient to isolate the green beam spots, but needs to be improved for carrying out a better centroiding, which for now causes too high errors. In the first experiment, it was noticed that the distance error was increasing when the object was moved away from the calibration position. In order to reduce this error, the lens distortion was taken into account thanks to the Matlab Calibration Toolbox. As the results were slightly better, it could be concluded they do not need to be considered. However, the last value, far from the reference, is confusing and the distortion needs to be corrected. Then, even if the results of the experiment with the line of dots show that the algorithm is robust but would not be rigorous enough to obtain a precise 3D map of the target, it gives us good prospects. Indeed, the poor calibration suggests that the algorithm could be much more errorless with an accurate calibration carried out with adapted tools which we did not have.

As regards the real time, as the system has to rectify the position of the rover's arm in order to keep the camera focused on the target, the algorithm needs to be fast enough to send as quickly as possible the data needed for the revision of the position. This goal is

also met as the algorithms work in real time and permit to receive the different distances continuously.

Finally, we manage to design an embedded system gathering a camera, an artificial light source and algorithms, which seems able to carry out a partial 3D map of a Martian rocks in real time, relevant for stabilizing the robot arm.

# Future Work

Although the theory section and the implementation of algorithms give a solid basis to achieve the goal of the system and the experiments are promising, there is still work to do.

The first step would be carrying out the different experiments once again but with adapted tools in order to obtain a precise calibration. Moreover, the color detection algorithm could be improved in order to detect the same pixels over time. Once these two improvements done, it would be interesting to compute the new errors of the experiments to check whether or not they decrease a lot as expected and if these errors are small enough to achieve an accurate determination of the distance board-target. It would also be appreciable to carry out integration tests with the conceived artificial light source and the camera. Indeed, even if the Signal/Noise ratio calculated validates their design, tests have to be performed. For instance, it could permit to verify the depth of field, the SNR and if the camera works well with a target between one and two meters. It could also allow to implement an algorithm to adjust the shutter time linked with the luminosity of the scene over time. Then, as the algorithms work with a line of lighting dots, it should also be efficient with a grid, and thus an experiment should be executed. Finally, an integration test should be carried out on the whole system, that is to say the camera, the artificial light source plus the algorithms.

Moreover, the use of patterns could be considered instead of a grid of dots. Indeed, the projection of a pattern could permit a more detailed 3D map. We could also manipulate a grid with more points into the same surface but a lot of pattern disorders would appear into the projected grid whereas patterns such as the ones described in the Theory Section permit to detect and correct these disorders.

Also, even if the system works well, we have to be sure that it will still be the case on Mars. To do so, it would be interesting to effectuate some researches on the resistance of the components. Indeed, as we said into the section 1.1.2, there are extreme variations in temperatures, large storms, wind and dust. Thus, the climate and the dust can damage the components or decrease their performance or even lead to malfunctions. For instance the lens of the camera striped by the dust would cause an erroneous detection of the lighting dots. Moreover, the calibration (namely the angles between the focal axis of the camera and the beams of the artificial light source) will probably change because of the vibrations

during the take-off and landing of the spacecraft. Thus, a method to allow the rover to calibrate the system itself should be studied.

Once the determination of the distances camera-target will give precise results, two algorithms will need to be implemented. The first one will compute the translations and rotations between two successive 3D maps and the second one will use these results to find the parameters needed to stabilize the camera of the rover.

Finally, we need to embed the system. The mechanical and electronic issues will need to be solved. For example, the different algorithms must be embedded on processor(s). In order to choose the CPU, the issues of the computation power (the processor needs to be powerful enough to have quick run time to work in real time) and the power consumption (there is a limited power available) will need to be balanced to find an efficient compromise. Also, as the artificial light source will probably not be on board at the same level as the camera, different distances  $d$  will have to be measured during the calibration. Indeed, as the artificial light source will probably be behind the camera, the distances  $d$  must be measured between the camera and each beam of the light source.

# Appendices

## .1 Results of the Experiment With a Line of Dots

Table 4: Object in front of dot 9 with distance object-board = 0.50 m (without noise)

i	$z_i$ [m]	$z_i$ error <sup>1</sup> [%]	distance board-Object [m]	distance board-object error <sup>1</sup> [%]
1	5.16205	0.92	0.048	
2	5.18899	0.40	0.021	
3	5.20282	0.14	0.007	
4	5.20879	0.02	0.001	
5	5.20999	0.00	0.000	
6	5.17524	0.67	0.035	
7	5.21324	0.06	-0.003	
8	5.22243	0.24	-0.012	
9	4.75529	0.01	0.455	9.06
10	5.27336	1.22	-0.063	

---

<sup>1</sup>error =  $\frac{distance_{theory} - distance_{experiment}}{distance_{theory}} 100$

Table 5: Object in front of dots 1 and 2 with distance object-board = 0.90 m (without noise)

i	$z_i$ [m]	$z_i$ error [%]	distance board-Object [m]	distance board-object error [%]
1	4.34553	0.82	0.864	3.95
2	4.36417	1.26	0.846	6.02
3	5.23825	0.54	-0.028	
4	5.2443	0.66	-0.034	
5	5.20999	0.00	0.000	
6	5.17524	0,67	0,035	
7	5.21324	0,06	-0,003	
8	5.22243	0,24	-0,012	
9	5.27727	1,29	-0,067	
10	5.23746	0,53	-0,027	

Table 6: Object in front of dots 6 and 7 with distance object-board = 0.70 m and noise

i	$z_i$ [m]	$z_i$ error [%]	distance board-Object [m]	distance board-object error [%]
1	5.16204	0.92	0.048	
2	5.18898	0.40	0.021	
3	5.20312	0.13	0.007	
4	5.20879	0.02	0.001	
5	5.20999	0.00	0.000	
6	4.53458	0.55	0.675	3.51
7	4.51034	0.01	0.700	0.04
8	5.22242	0.24	-0.012	
9	5.24131	0.60	-0.031	
10	5.30848	1.89	-0.09848	

## **.2 LED Properties**

### **Caractéristiques**

• Couleur	Vert
• Température de couleur	525nm
• Intensité Lumineuse	84000mcd
• Ampérage Requis	20mA
• Angle de Diffusion	30°
• Consommation	220mW
• Tension d'alimentation	9.3 - 10.5V
• Durée de vie estimée	80 000 Heures

Figure 9: LED Datasheet

### 3 Matlab Camera Calibrator Toolbox

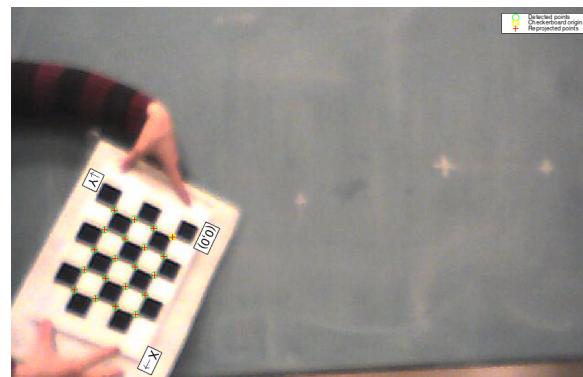


Figure 10: Detection of the Checkerboard

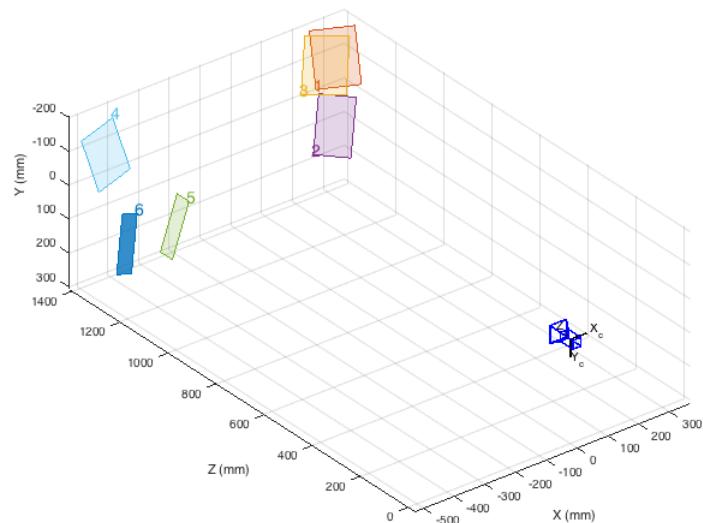
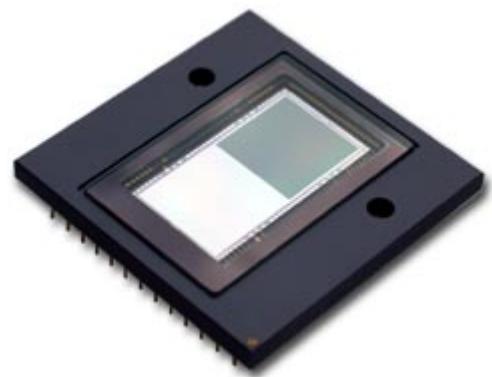


Figure 11: Distance between the different positions of the checkerboard and the camera

## **.4 CCD Properties**

# DATA SHEET



## **FTT1010M** 1M Frame Transfer CCD Image Sensor

Product specification

2007, April 17

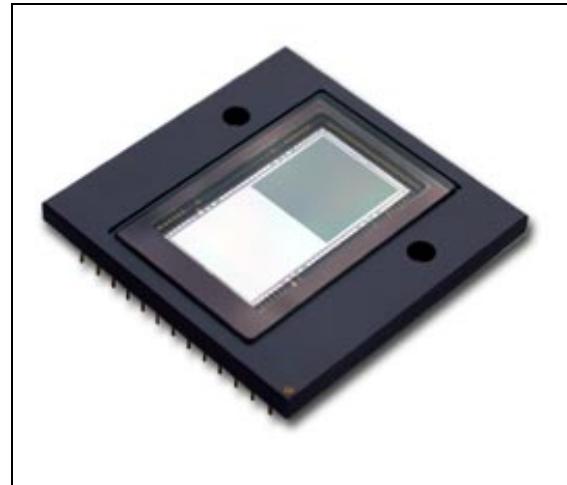
**DALSA**  
Professional Imaging

**DALSA**

## 1M Frame Transfer CCD Image Sensor

FTT1010M

- **1-inch optical format**
- **1M active pixels (1024H x 1024V)**
- **Progressive scan**
- **Excellent antiblooming**
- **Variable electronic shuttering**
- **Square pixel structure**
- **H and V binning**
- **100% fill factor**
- **High dynamic range (>72dB)**
- **High sensitivity**
- **Low dark current and fixed pattern noise**
- **Low readout noise**
- **Data rate up to 2 x 40 MHz**
- **Mirrored and split readout**
- **RoHS compliant**

**Description**

The FTT1010M is a monochrome progressive-scan frame-transfer image sensor offering 1K x 1K pixels at 30 frames per second through a single output buffer. The combination of high speed and a high linear dynamic range (>12 true bits at room temperature without cooling) makes this device the perfect solution for high-end real time medical X-ray, scientific and industrial applications. A second output can either be used for mirrored images, or can be read out simultaneously with other output to double the frame rate. The device structure is shown in figure 1.

**Device structure**

Optical size: 12.288 mm (H) x 12.288 mm (V)  
 Chip size: 14.572 mm (H) x 26.508 mm (V)  
 Pixel size: 12  $\mu$ m x 12  $\mu$ m  
 Active pixels: 1024 (H) x 1024 (V)  
 Total no. of pixels: 1072 (H) x 1030 (V)  
 Optical black pixels: Left: 20 Right: 20  
 Timing pixels: Left: 4 Right: 4  
 Dummy register pixels: Left: 7 Right: 7  
 Optical black lines: Bottom: 6 Top: 6

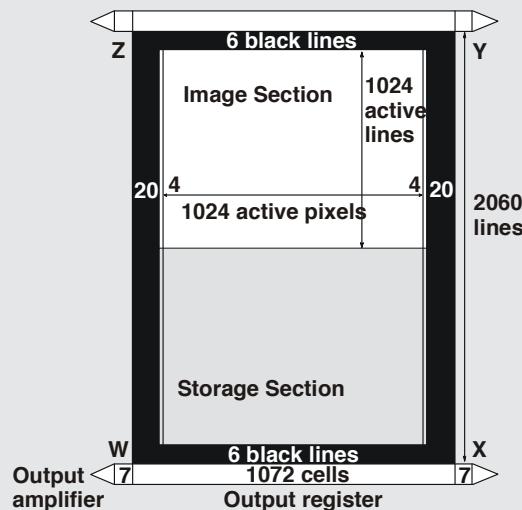


Figure 1 – Device Structure

## 1M Frame Transfer CCD Image Sensor

FTT1010M

## Architecture of the FTT1010M

The FTT1010M consists of a shielded storage section and an open image section. Both sections are electronically the same and have the same cell structure with the same properties. The only difference between two sections is the optical light shield.

The optical centres of all pixels in the image section form a square grid. The charge is generated and integrated in this section. Output registers are located below the storage section. The output amplifiers Y and Z are not used in Frame Transfer mode and should be connected as not-used amplifiers.

After the integration time, the charge collected in the image section is shifted to the storage section. The charge is read out line by line through the lower output register.

The left and the right half of each output register can be controlled independently. This enables either single or multiple readout.

During vertical transport, the C3 gates separate the pixels in the register. The letters W, X, Y, and Z are used to define the four quadrants of the sensor. The central C3 gates of both registers are part of the W and Z quadrants of the sensor.

Both upper and lower registers can be used for vertical binning. Both registers also have a summing gate at each end that can be used for horizontal binning. Figure 2 shows the detailed internal structure.

IMAGE SECTION	
Image diagonal (active video only)	17.38 mm
Aspect ratio	1:1
Active image width x height	12.288 x 12.288 mm <sup>2</sup>
Pixel width x height	12 x 12 $\mu\text{m}^2$
Fill factor	100%
Image clock pins	A1, A2, A3, A4
Capacity of each clock phase	2.5nF per pin
Number of active lines	1024
Number of black reference lines	6
Total number of lines	1030
Number of active pixels per line	1024
Number of overscan (timing) pixels per line	8 (2x4)
Number of black reference pixels per line	40 (2x20)
Total number of pixels per line	1072

STORAGE SECTION	
Storage width x height	12.864 x 12.360 mm <sup>2</sup>
Cell width x height	12 x 12 $\mu\text{m}^2$
Storage clock phases	B1, B2, B3, B4
Capacity of each clock phase	2.5nF per pin
Number of cells per line	1072
Number of lines	1030

OUTPUT REGISTERS	
Output buffers (three-stage source follower)	4 (one on each corner)
Number of registers	2 (one above, one below)
Number of dummy cells per register	14 (2x7)
Number of register cells per register	1072
Output register horizontal transport clock pins	C1, C2, C3
Capacity of each C-clock phase	60pF per pin
Overlap capacity between neighbouring C-clocks	20pF
Output register Summing Gates	4 pins (SG)
Capacity of each SG	15pF
Reset Gate clock phases	4 pins (RG)
Capacity of each RG	15pF

## 1M Frame Transfer CCD Image Sensor

FTT1010M

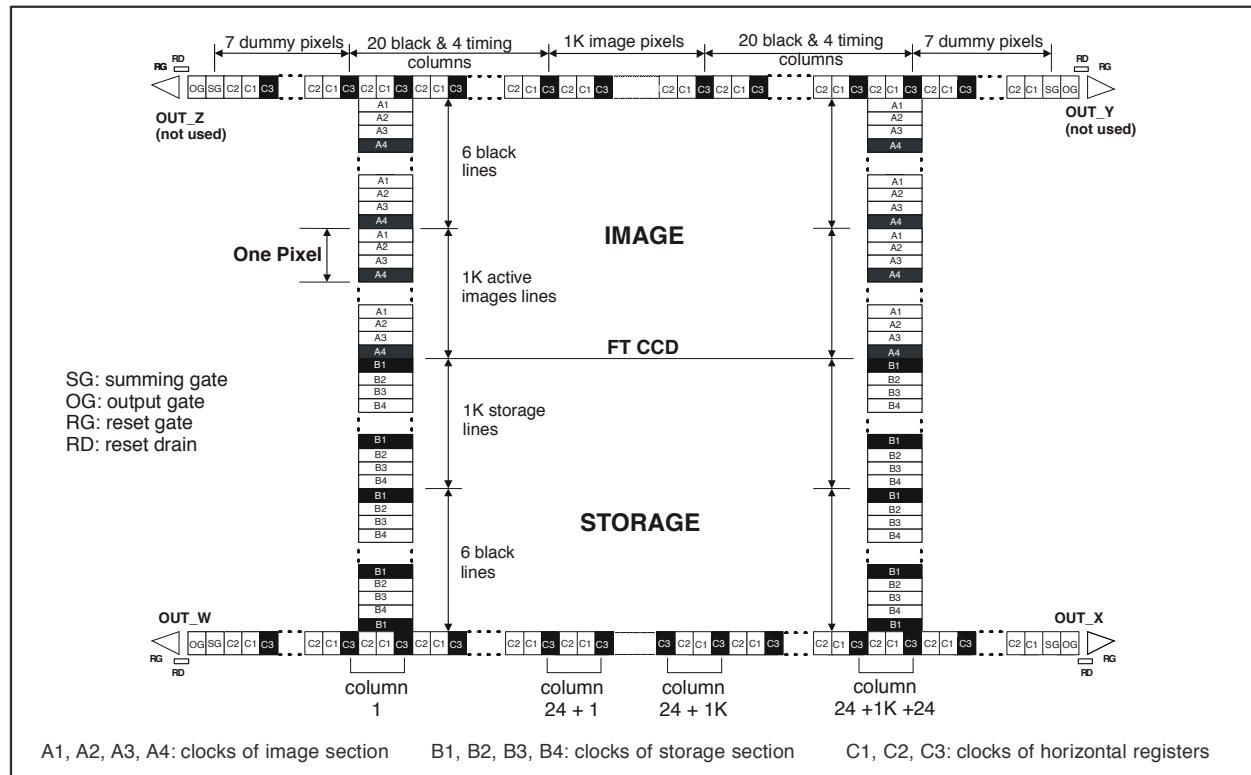


Figure 2 - Detailed internal structure

## 1M Frame Transfer CCD Image Sensor

FTT1010M

## Specifications

ABSOLUTE MAXIMUM RATINGS <sup>1</sup>		MIN	MAX	UNIT
GENERAL:				
storage temperature		-55	+80	°C
ambient temperature during operation		-40	+60	°C
voltage between any two gates		-20	+20	V
DC current through any clock (absolute value)		-0.2	+0.2	µA
OUT current (no short circuit protection)		0	10	mA
VOLTAGES IN RELATION TO VPS:				
VPS, SFD, RD		-0.5	+30	V
VCS, SFS		-8	+5	V
All other pins		-5	+25	V
VOLTAGES IN RELATION TO VNS:				
SFD, RD		-15	+0.5	V
VCS, SFS, VPS		-30	+0.5	V
All other pins		-30	+0.5	V
DC CONDITIONS <sup>2,3</sup>		MIN [V]	TYPICAL [V]	MAX [V]
VNS <sup>4</sup>	N substrate	20	24	28
VPS	P substrate	1	3	7
SFD	Source Follower Drain	16	20	24
SFS	Source Follower Source	-	0	-
VCS	Current Source	-5	0	3
OG	Output Gate	4	6	8
RD	Reset Drain	13	15.5	18
AC CLOCK LEVEL CONDITIONS <sup>2</sup>		MIN	TYPICAL	MAX
IMAGECLOCKS:				
A-clock amplitude during integration and hold		8	10	V
A-clock amplitude during vertical transport (duty cycle=5/8) <sup>5</sup>		10	14	V
A-clock low level			0	V
Charge Reset (CR) level on A-clock <sup>6</sup>		-5	-5	V
STORAGE CLOCKS:				
B-clock amplitude during hold		8	10	V
B-clock amplitude during vertical transport (duty cycle=5/8)		10	14	V
OUTPUT REGISTER CLOCKS:				
C-clock amplitude (duty cycle during hor. transport=3/6)		4.75	5	5.25
C-clock low level		2	3.5	V
Summing Gate (SG) amplitude			10	V
Summing Gate (SG) low level			3.5	V
OTHER CLOCKS:				
Reset Gate (RG) amplitude		5	10	V
Reset Gate (RG) low level			3	V
Charge Reset (CR) pulse on Nsub <sup>6</sup>		0	10	V

<sup>1</sup> During Charge Reset it is allowed to exceed maximum rating levels (see note 5)<sup>2</sup> All voltages in relation to SFS<sup>3</sup> Power-up sequence: VNS, SFD, RD, VPS, others<sup>4</sup> To set the VNS voltage for optimal Vertical Antiblooming (VAB), it should be adjustable between minimum and maximum values<sup>5</sup> Three-level clock is preferred for maximum charge; the swing during vertical transport should be 4V higher than the voltage during integration<sup>6</sup> A two level clock (typically 10V) can be used if a lower maximum charge handling capacity is allowed<sup>6</sup> Charge Reset can be achieved in two ways:

- The typical CR level is applied to all image clocks simultaneously (preferred).
- The typical A-clock low level is applied to all image clocks; for proper CR, an additional Charge Reset pulse on VNS is required. This will also affect the charge handling capacity in the storage areas.

## 1M Frame Transfer CCD Image Sensor

FTT1010M

## Timing diagrams (for default operation)

AC CHARACTERISTICS	MIN	TYPICAL	MAX	UNIT
Horizontal frequency (1/Tp) <sup>1</sup>	0	18	40	MHz
Vertical frequency	0	450	1000	kHz
Charge Reset (CR) time	2	20		μs
Rise and fall times: image clocks (A)	10	20		ns
storage clocks (B)	10	20		ns
register clocks (C) <sup>2</sup>	3	5	1/6 Tp	ns
summing gate (SG)	3	5	1/6 Tp	ns
reset gate (RG)	3	5	1/6 Tp	ns

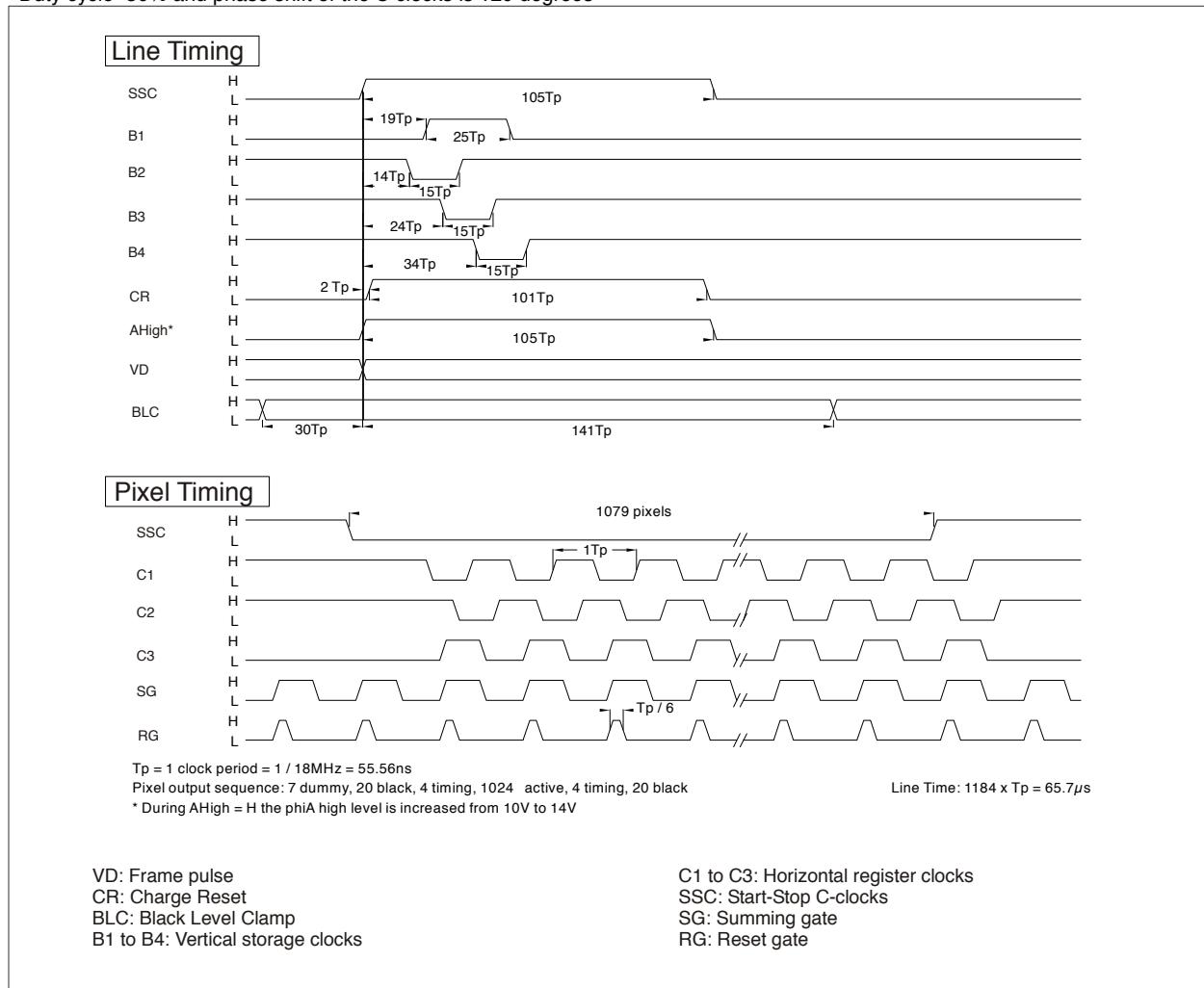
<sup>1</sup>TP=1 clock period<sup>2</sup>Duty cycle=50% and phase shift of the C clocks is 120 degrees

Figure 3 - Line and pixel timing diagrams

## 1M Frame Transfer CCD Image Sensor

FTT1010M

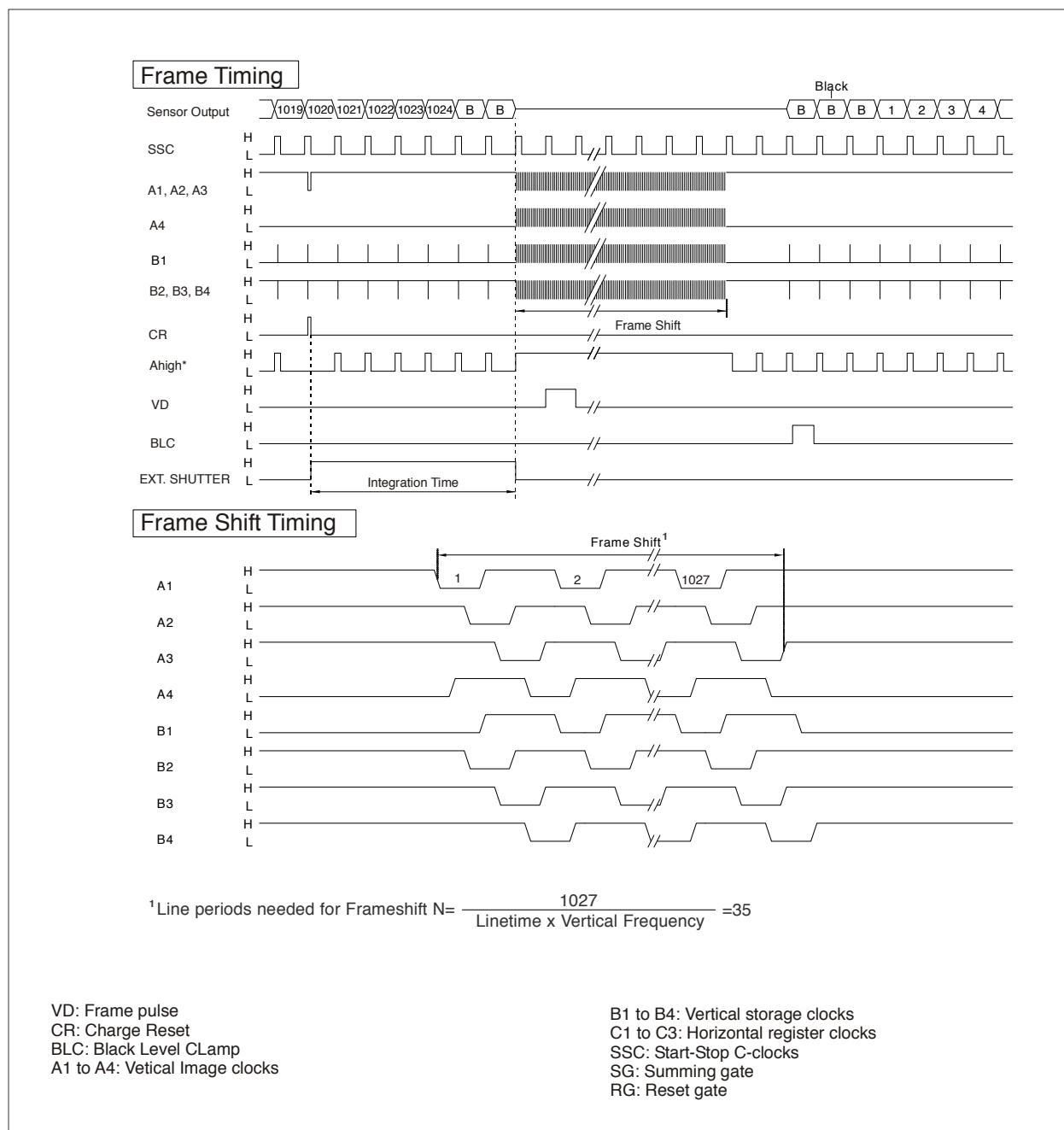


Figure 4 - Frame timing diagrams

## 1M Frame Transfer CCD Image Sensor

FTT1010M

## Line timing

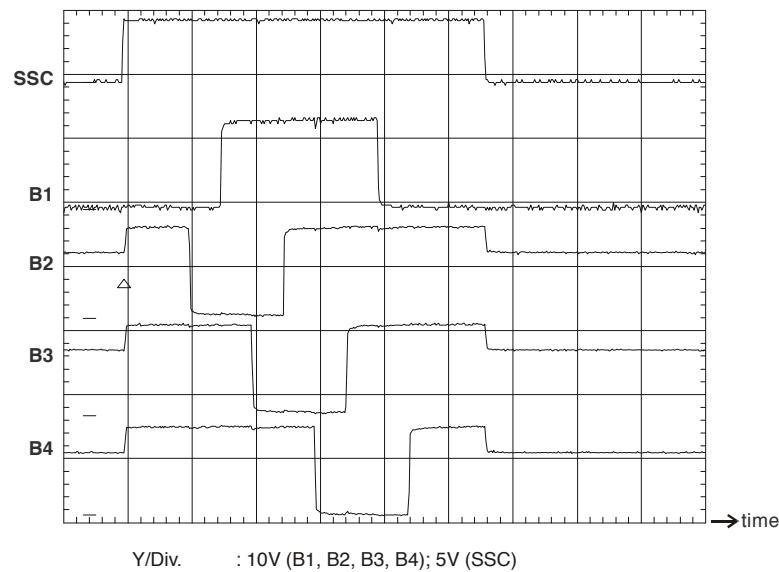


Figure 5 - Vertical readout

## Pixel timing

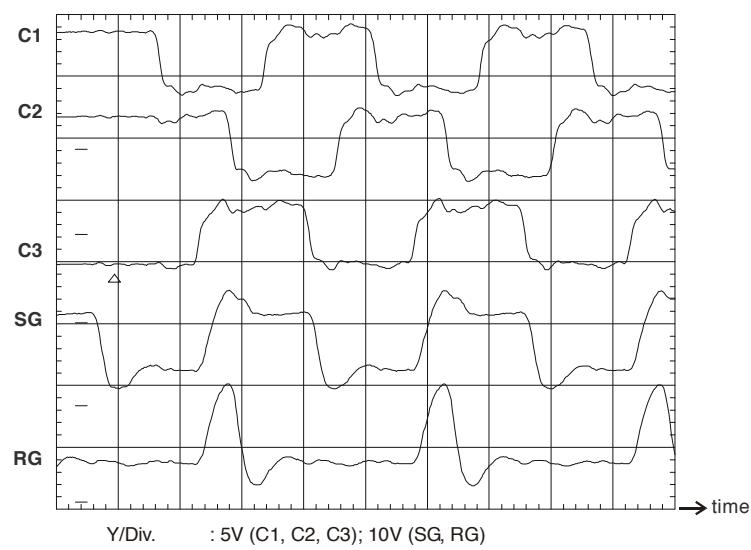


Figure 6 - Start horizontal readout

## 1M Frame Transfer CCD Image Sensor

FTT1010M

**Performance**

The test conditions for the performance characteristics are as follows:

- All values are measured using typical operating conditions.
- VNS is adjusted as low as possible while maintaining proper Vertical Antiblooming
- Sensor temperature=60°C (333K)
- Horizontal transport frequency=18MHz
- Vertical transport frequency=450kHz
- Integration time=10ms
- The light source is a lamp of 3200K in conjunction with neutral density filters and a 1.7mm thick BG40 infrared cut-off filter. For Linear Operation measurements, a temperature conversion filter (Melles Griot type no. 03FCG261, -120 mired, thickness: 2.5mm) is applied.

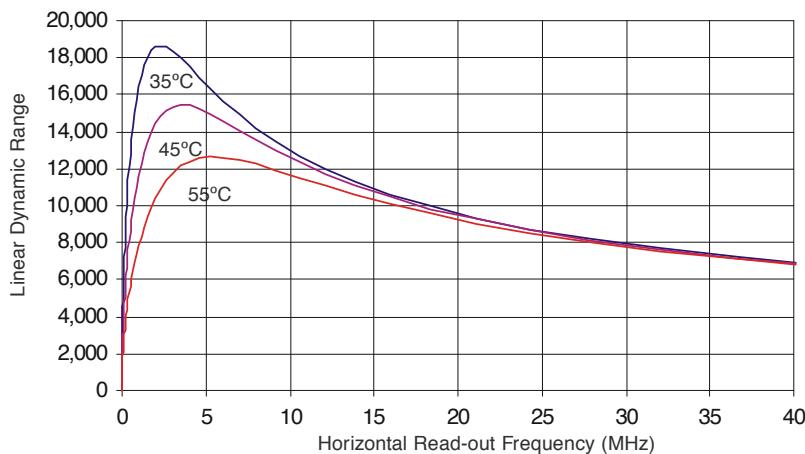
LINEAR OPERATION	MIN	TYPICAL	MAX	UNIT
Charge Transfer Efficiency <sup>1</sup> vertical		0.999995		
Charge Transfer Efficiency <sup>1</sup> horizontal		0.999999		
Image lag			0	%
Smear <sup>2</sup>		-39	0	dB
Resolution (MTF) @ 42lp/mm	65	250		%
Light sensitivity	180			kel/lux•s
Low Pass Shading <sup>3</sup>			2.5	%
Random Non-Uniformity (RNU) <sup>4</sup>		0.3	5	%

<sup>1</sup>Charge Transfer Efficiency values are tested by evaluation and expressed as the value per gate transfer.

<sup>2</sup>Smear is defined as the ratio of 10% of the vertical transport time to the integration time. It indicates how visible a spot of 10% of the image height would become.

<sup>3</sup>Low Pass Shading is defined as the ratio of the one- $\sigma$  value of the pixel output distribution expressed as a percentage of the mean value output (low-pass image).

<sup>4</sup>RNU is defined as the ratio of the one- $\sigma$  value of the highpass image to the mean signal value at nominal light.



Linear dynamic range is defined as the ratio of  $Q_{lin}$  to read-out noise (the latter reduced by Correlated Double Sampling).

Figure 7 - Typical Linear dynamic range vs. horizontal read-out frequency and sensor temperature

## 1M Frame Transfer CCD Image Sensor

FTT1010M

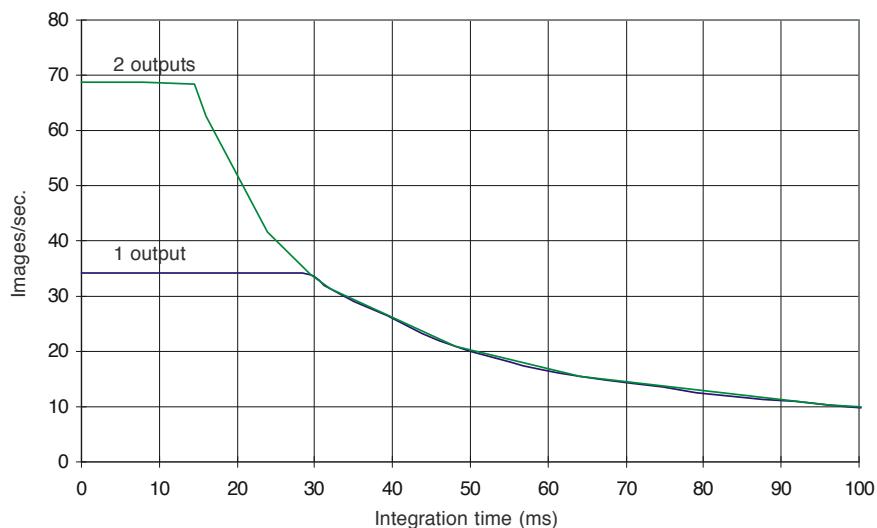


Figure 8 - Maximum number of images/second versus integration time

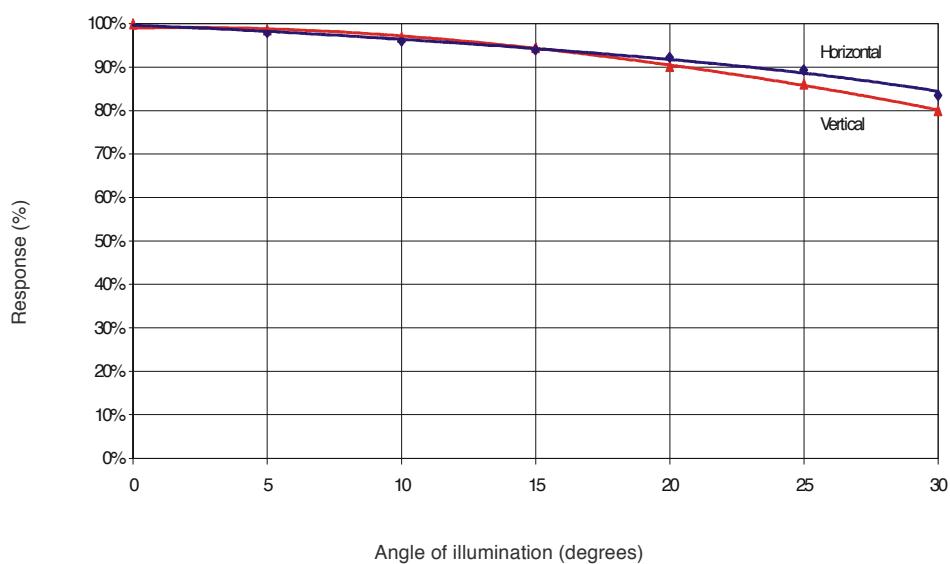


Figure 9 - Angular response versus angle of illumination

## 1M Frame Transfer CCD Image Sensor

FTT1010M

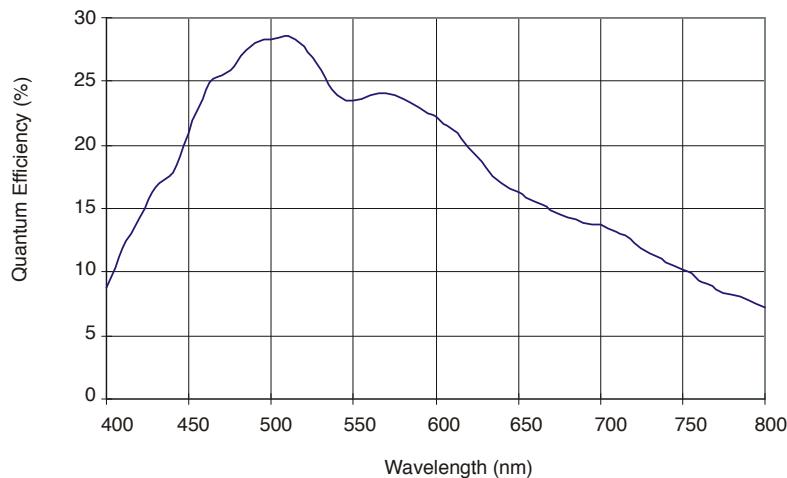


Figure 10 - Quantum efficiency versus wavelength

## 1M Frame Transfer CCD Image Sensor

FTT1010M

LINEAR/SATURATION	MIN	TYPICAL	MAX	UNIT
Full-well capacity saturation level (Qmax) <sup>1</sup>	250	500	600	kel
Full-well capacity linear operation (Qlin) <sup>2</sup>	200	350		kel
Charge handling capacity <sup>3</sup>		600		kel
Overexposure <sup>4</sup> handling	100	200		x Qmax level

<sup>1</sup>Qmax is determined from the low-pass filtered image.

<sup>2</sup>The linear full-well capacity Qlin is calculated from linearity test (see dynamic range). The evaluation test guarantees 97% linearity.

<sup>3</sup>Charge handling capacity is the largest charge packet that can be transported through the register and read-out through the output buffer.

<sup>4</sup>Overexposure over entire area while maintaining good Vertical Antiblooming (VAB). It is tested by measuring the dark line.

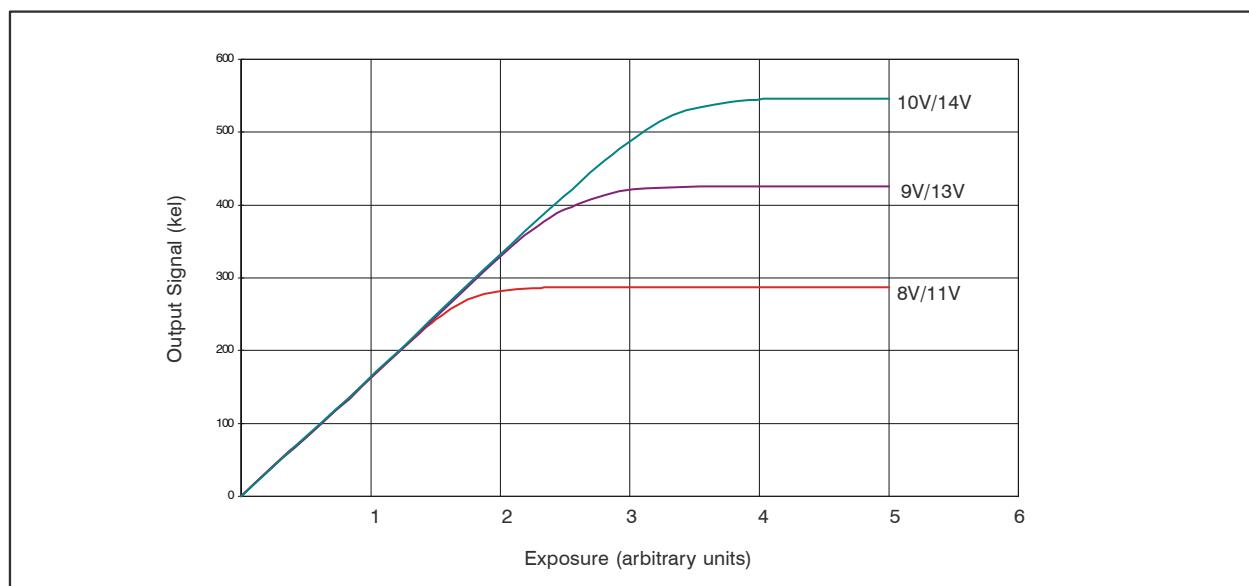


Figure 11 – Charge handling versus integration/transport voltage

## 1M Frame Transfer CCD Image Sensor

FTT1010M

OUTPUT BUFFERS	MIN	TYPICAL	MAX	UNIT
Conversion factor	6	8	12	$\mu\text{V/el}$
Mutual conversion factor matching ( $\Delta\text{ACF}$ ) <sup>1</sup>		0	2	$\mu\text{V/el}$
Supply current		4		mA
Bandwidth		110		MHz
Output impedance buffer ( $R_{\text{load}}=3.3\Omega$ , $C_{\text{load}}=2\text{pF}$ )		400		$\Omega$

<sup>1</sup>Matching of the four outputs is specified as  $\Delta\text{ACF}$  with respect to reference measured at the operating point ( $Q_{\text{in}}/2$ )

DARK CONDITION	MIN	TYPICAL	MAX	UNIT
Dark current level @ 20°C		10	30	$\text{pA/cm}^2$
Dark current level @ 60°C		0.3	0.6	$\text{nA/cm}^2$
Fixed Pattern Noise <sup>1</sup> (FPN) @ 60°C		15	25	el
RMS readout noise @ 9MHz bandwidth after CDS		25	30	el

<sup>1</sup>FPN is one- $\sigma$  value of the high-pass image.

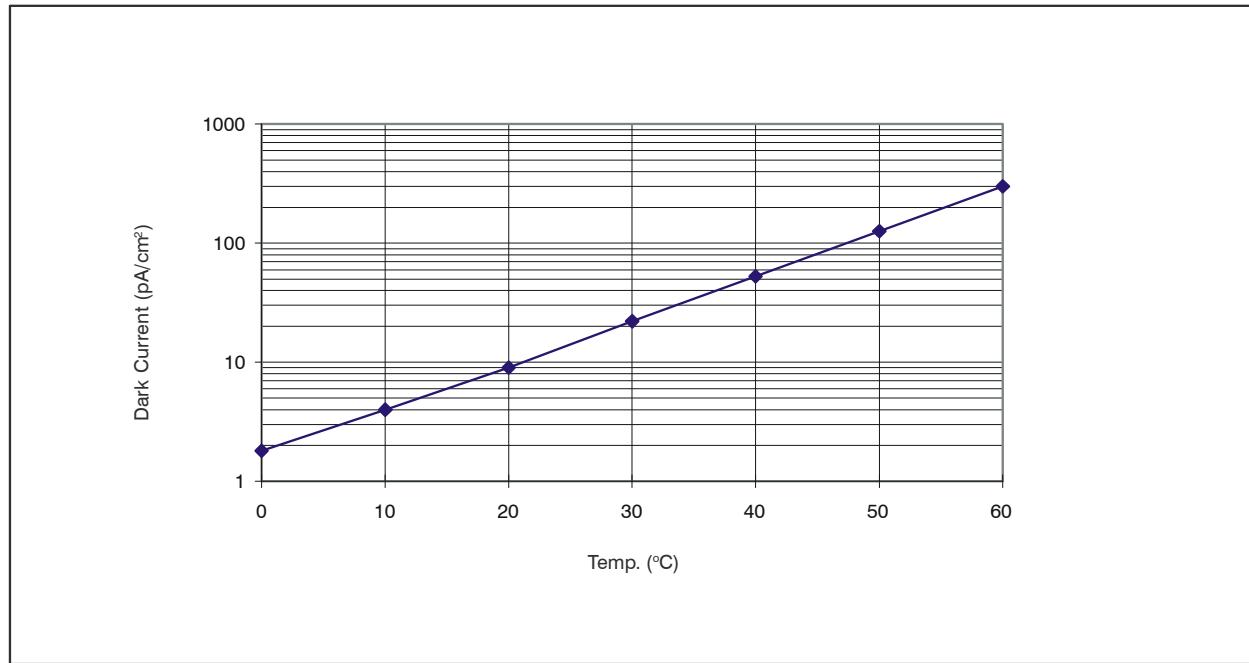


Figure 12 – Dark current versus temperature

# 1M Frame Transfer CCD Image Sensor

FTT1010M

## Application information

### *Current handling*

One of the purposes of VPS is to drain the holes that are generated during exposure of the sensor to light. Free electrons are either transported to the VRD connection and, if excessive (from overexposure), free electrons are drained to VNS. No current should flow into any VPS connection of the sensor. During high overexposure a total current of 10 to 15mA through all VPS connections together may be expected. The PNP emitter follower in the circuit diagram (figure 12) serves these current requirements.

VNS drains superfluous electrons as a result of overexposure. In other words, it only sinks current. During high overexposure, a total current of 10 to 15mA through all VNS connections together may be expected. The clamp circuit, consisting of the diode and electrolytic capacitor, enables the addition of a Charge Reset (CR) pulse on top of an otherwise stable VNS voltage. To protect the CCD, the current resulting from this pulse should be limited. This can be accomplished by designing a pulse generator with a rather high output impedance.

### *Decoupling of DC voltages*

All DC voltages (not VNS, which has additional CR pulses as described above) should be decoupled with a 100nF decoupling capacitor. This capacitor must be mounted as close as possible to the sensor pin. Further noise reduction (by bandwidth limiting) is achieved by the resistors in the connections between the sensor and its voltage supplies. The electrons that build up the charge packets that will reach the floating diffusions only add up to a small current, which will float through VRD. Therefore, a large series resistor in the VRD connection may be used.

### *Outputs*

To limit the on-chip power dissipation, the output buffers are designed with open source outputs. Outputs to be used should therefore be loaded with a current source or more simply with a resistance to GND. In order to prevent the output (which typically has an output impedance of about  $400\Omega$ ) from bandwidth limitation as a result of capacitive loading, load the output with an emitter follower built from a high-frequency transistor. Mount the base of this transistor as close as possible to the sensor and keep the connection between the emitter and the next stage short. The CCD output buffer can easily be destroyed by ESD. By using this

emitter follower, this danger is suppressed; do NOT reintroduce this danger by measuring directly on the output pin of the sensor with an oscilloscope probe. Instead, measure on the output of the emitter follower. Slew rate limitation is avoided by avoiding a too-small quiescent current in the emitter follower; about 10mA should do the job. The collector of the emitter follower should be decoupled properly to suppress the Miller effect from the base-collector capacitance.

A CCD output load resistor of  $3.3\Omega$  typically results in a bandwidth of 110MHz. The bandwidth can be enlarged to about 130MHz by using a resistor of  $2.2k\Omega$  instead, which, however, also enlarges the on-chip power dissipation.

### *Device protection*

The output buffers of the FTT1010M are likely to be damaged if VPS rises above SFD or RD at any time. This danger is most realistic during power-on or power-off of the camera. The RD voltage should always be lower than the SFD voltage.

Never exceed the maximum output current. This may damage the device permanently. The maximum output current should be limited to 10mA. Be especially aware that the output buffers of these image sensors are very sensitive to ESD damage.

Because of the fact that our CCDs are built on an n-type substrate, we are dealing with some parasitic npn transistors. To avoid activation of these transistors during switch-on and switch-off of the camera, we recommend the application diagram of figure 12.

### *Unused sections*

To reduce power consumption, the following steps can be taken. Connect unused output register pins (C1...C3, SG, OG) and unused SFS pins to zero Volts.

### *More information*

Detailed application information is provided in the application note AN01 entitled **“Camera Electronics for the mK x nK CCD Image Sensor Family”**.

## 1M Frame Transfer CCD Image Sensor

FTT1010M

## Device Handling

An image sensor is a MOS device which can be destroyed by electro-static discharge (ESD). Therefore, the device should be handled with care.

Always store the device with short-circuiting clamps or on conductive foam. Always switch off all electric signals when inserting or removing the sensor into or from a camera (the ESD protection in the CCD image sensor process is less effective than the ESD protection of standard CMOS circuits).

Being a high quality optical device, it is important that the cover glass remain undamaged. When handling the sensor, use fingercots.

When cleaning the glass we recommend using ethanol (or possibly water). Use of other liquids is strongly discouraged:

- if the cleaning liquid evaporates too quickly, rubbing is likely to cause ESD damage.
- the cover glass and its coating can be damaged by other liquids.

Rub the window carefully and slowly.

Dry rubbing of the window may cause electro-static charges or scratches which can destroy the device.

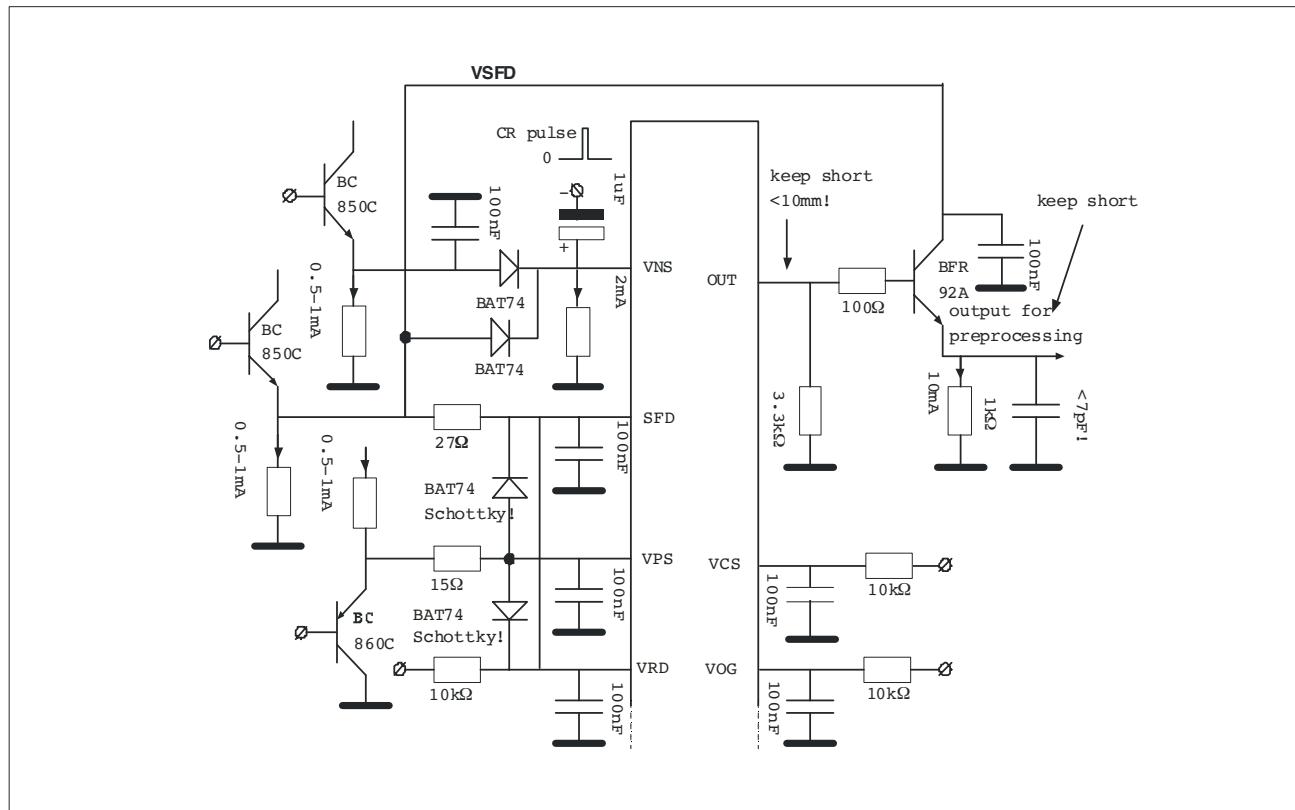


Figure 13- Application diagram

## 1M Frame Transfer CCD Image Sensor

FTT1010M

## Pin configuration

The FTT1010M is mounted in a Pin Grid Array (PGA) package with 80 pins in a 15x13 grid of 40.00 x 40.00 mm<sup>2</sup>. The position of pin A1 is marked with a gold dot on top of the

package. The image clock phases of quadrant W are internally connected to X, and the clock phases of Y are connected to Z.

SYMBOL	LINEAR/SATURATION	PIN # W	PIN # X	PIN # Y	PIN # Z
VNS	N substrate	A12	A3	J2	F11
VNS	N substrate	D11	B2	F3	H12
VNS	N substrate	E11	D3	-	J11
VNS	N substrate	E12	E2	-	-
VNS	N substrate	-	E3	-	-
VPS	P substrate	C11	C3	G3	G11
SFD	Source Follower Drain	A13	A1	J1	J13
SFS	Source Follower Source	A10	B5	J4	H9
VCS	Current Source	A11	A4	J3	J10
OG	Output Gate	B13	B1	H1	H13
RD	Reset Drain	B12	B3	H2	H11
A1	Image Clock (Phase 1)	-	-	F1	F13
A2	Image Clock (Phase 2)	-	-	G2	G12
A3	Image Clock (Phase 3)	-	-	F2	F12
A4	Image Clock (Phase 4)	-	-	G1	G13
B1	Storage Clock (Phase 1)	D13	D1	-	-
B2	Storage Clock (Phase 2)	C12	C2	-	-
B3	Storage Clock (Phase 3)	D12	D2	-	-
B4	Storage Clock (Phase 4)	C13	C1	-	-
C1	Register Clock (Phase 1)	B9	A6	H5	J8
C2	Register Clock (Phase 2)	B8	A7	H6	J7
C3	Register Clock (Phase 3)	A8	B6	J6	H8
SG	Summing Gate	B10	A2	H4	J12
RG	Reset Gate	A9	A5	J5	J9
OUT	Output	B11	B4	H3	H10
NC	Not Connected	B7	-	H7	-

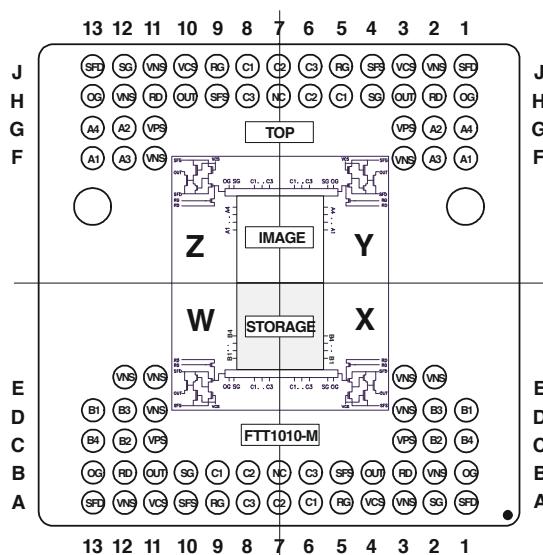


Figure 14 - Pin configuration (top view)

## 1M Frame Transfer CCD Image Sensor

FTT1010M

## Package information

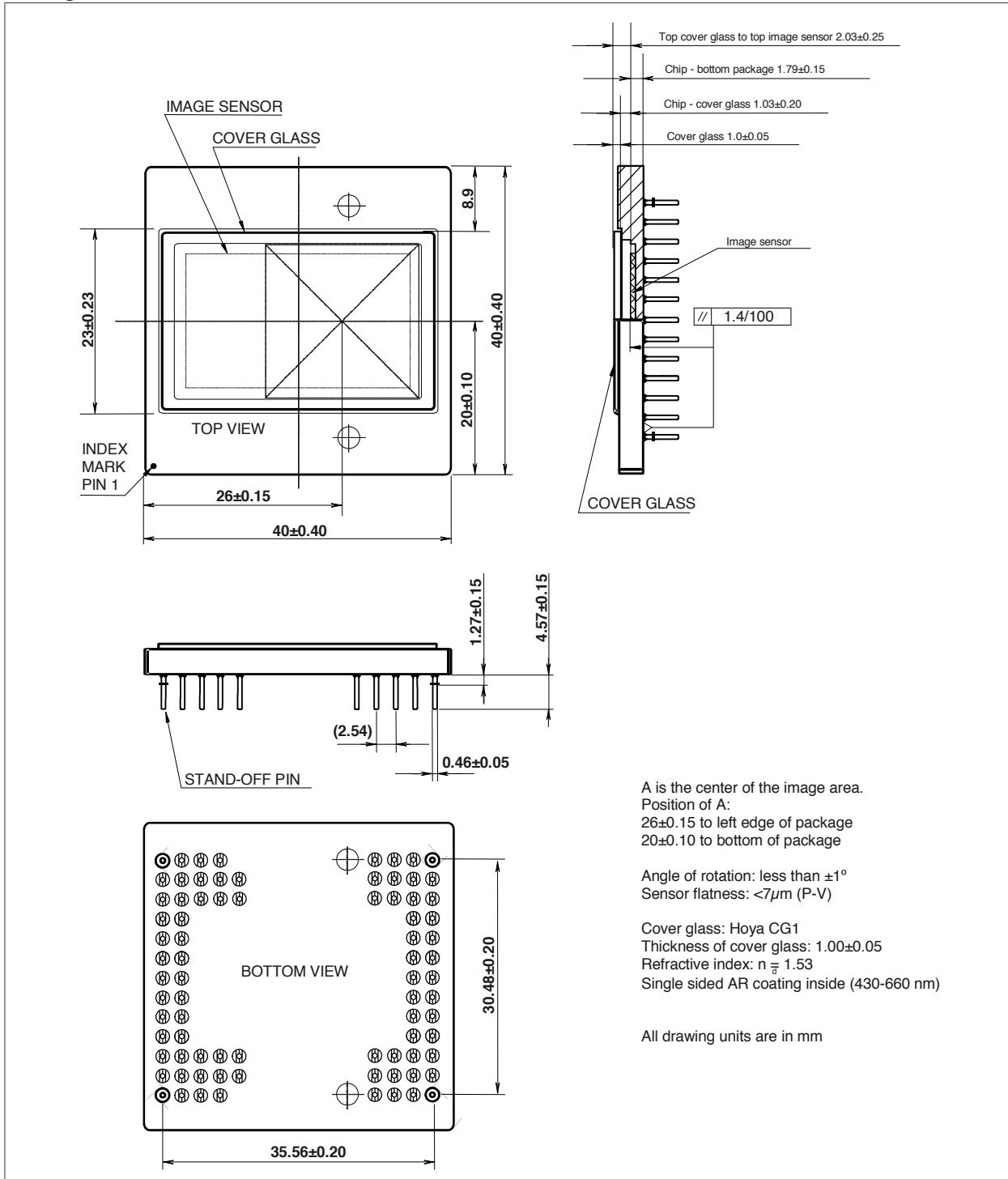


Figure 15—Mechanical drawing of the PGA package of the FTT1010M

**1M Frame Transfer CCD Image Sensor****FTT1010M****Order codes**

The sensor can be ordered using the following code:

<b>FTT1010M sensors</b>		
<b>Description</b>	<b>Quality Grade</b>	<b>Order Code</b>
FTT1010M/TG	Test grade	9922 157 35231
FTT1010M/EG	Economy grade	9922 157 35251
FTT1010M/IG	Industrial grade	9922 157 35221
FTT1010M/HG	High grade	9922 157 35211

**Defect Specifications**

The CCD image sensor can be ordered in a specific quality grade. The grading is defined with the maximum amount of pixel defects, column defects, row defects and cluster defects, in both illuminated and non-illuminated conditions. For detailed grading information, please contact your local DALSA representative.

**For More Information**

For more detailed information on this and other products, contact your local rep or visit our Web site at  
<http://www.dalsa.com/sensors/products/products.asp>.

**DALSA Professional Imaging**

Sales Department  
 High Tech Campus 27  
 5656 AE Eindhoven  
 The Netherlands  
 Tel: +31 40 259 9009  
 Fax: +31 40 259 9015  
[www.dalsa.com/sensors](http://www.dalsa.com/sensors)  
[sales.sensors@dalsa.com](mailto:sales.sensors@dalsa.com)

**This information is subject to change without notice.**



# Bibliography

- [1] Hubblesite. <http://hubblesite.org/newscenter/archive/releases/2001/31/image/a/>. Accessed: December 17, 2015.
- [2] Nasa mars orbiter catches twister in action. <http://mars.nasa.gov/mro/news/whatsnew/index.cfm?FuseAction>ShowNews&NewsID=1210>. Accessed: December 17, 2015.
- [3] Gary Bradski and Adrian Kaehler. *Learning OpenCV: Computer vision with the OpenCV library*. " O'Reilly Media, Inc.", 2008.
- [4] Jason Geng. Structured-light 3d surface imaging: a tutorial. *Advances in Optics and Photonics*, 3(2):128–160, 2011.
- [5] Berthold Horn. *Robot vision*. MIT press, 1986.
- [6] John Leif Jørgensen, 2015.
- [7] Hubert H Lamb. Climate present past and future vol-1. 1975.
- [8] JN Maki, JF Bell, KE Herkenhoff, SW Squyres, A Kiely, M Klimesh, M Schwochert, T Litwin, R Willson, A Johnson, et al. Mars exploration rover engineering cameras. *Journal of Geophysical Research: Planets (1991–2012)*, 108(E12), 2003.
- [9] MathWorks. Computer vision system toolbox user's guide. pages 21–74, 2015.
- [10] Hiroyoshi Morita, Kaanyasn Yajima, and Shojiro Sakata. Reconstruction of surfaces of 3-d objects by m-array pattern projection method. In *Computer Vision., Second International Conference on*, pages 468–473. IEEE, 1988.
- [11] NASA. Robotic mars exploration, 2015. [Online; accessed December 17, 2015].
- [12] Giovanna Sansoni, Marco Trebeschi, and Franco Docchio. State-of-the-art and applications of 3d imaging sensors in industry, cultural heritage, medicine, and criminal investigation. *Sensors*, 9(1):568–601, 2009.

- [13] Wikipedia. Atmosphère de mars — Wikipedia, the free encyclopedia, 2015. [Online; accessed December 17, 2015].
- [14] Wikipedia. Circle of confusion — wikipedia, the free encyclopedia, 2015. [Online; accessed December 17, 2015].
- [15] Wikipedia. Hsl and hsv — wikipedia, the free encyclopedia, 2015. [Online; accessed December 17, 2015].
- [16] Giichi Yamamoto. Direct absorption of solar radiation by atmospheric water vapor, carbon dioxide and molecular oxygen. *Journal of the Atmospheric Sciences*, 19(2):182–188, 1962.