

# **4<sup>th</sup> year Internship report**

## **3D Map generation and autonomous navigation using Symbolic Computing**

**Synopsis**

**Handed over**  
**University of Petroleum and Energy Studies**

Carried out by  
**Maxime PINEAU**

Under the direction of  
**Doctor Niharika SINGH**

From 30th May 2016 to 26 August 2016

Version of  
June 5, 2016



UNIVERSITÉ DE NANTES

2015 - 2016

## Contents

<b>1</b>	<b>Abstract</b>	<b>1</b>
<b>2</b>	<b>Objectives</b>	<b>1</b>
<b>3</b>	<b>Introduction</b>	<b>1</b>
3.1	DEM generation . . . . .	2
3.2	Rajan Transformation and it's application in the sets . . . . .	2
3.2.1	Rajan Transformation . . . . .	2
3.2.2	Set Theoretic Rajan Transformation . . . . .	3
<b>4</b>	<b>Methodology</b>	<b>3</b>
	<b>References</b>	<b>3</b>
	<b>Glossary</b>	<b>3</b>

## 1 Abstract

This project involves moving an autonomous robot in an unknown environment, with a minimum of time and energy concupction. For example, the robot will have to explore and search into an unknown environment, such as a the planet Mars or a submarin field. An autonomus robot could be used in differents domains, such as surveillance (drones), transport, or cleaning. The robot will have to generate a 3D map of his surrounding, take a decision accordingly, move and then learn from this decision. The robot will use reinforcement learning to learn from his previous decisions. As for the generation of the 3D map, we will be using Digital Elevation Model (DEM) and Symbolic Computing.

**keywords :** 3D Map Generation, Autonomous Navigation, DEM, Symbolic Computing, STRT

## 2 Objectives

In order to generate the 3D map, we will use DEM, Symbolic Computing and a Cellular Logic Array Processing (CLAP) algorithm (which is used for image processing and pattern recognition). The DEM images will have to be transform with a Set Theoretic Rajan Transform (STRT), a tranformation involving Symbolic Computing. Nowadays 3D map generation technologies generally use DIP methods, but DEM method was selected because it is faster during the compilation time.

In order to visualize the effects of the differents techniques we will use, we will develop a Graphic User Interface (GUI) where each steps could be done separately. The user will have to charge an image and apply a transformation on it by selecting one. A new image will be generated, with the name of the operation and the time it has taken to process.

The reinforment learning algorithm, which will allow the robot to self-learning through experiences, will use a reward policy.

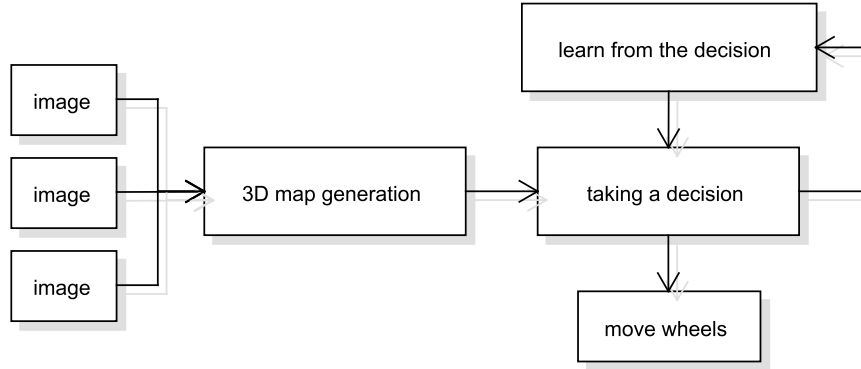
As the University doesn't have the actual robot for the moment, this project will be conduct under a simulated environnement.

## 3 Introduction

The robot will be equiped by differents camera, allowing it to take pictures from different angles. Those pictures, or input images, correspond to the surrounding of the robot. It will also be equiped with differents sensors that will provide additionnal information, such as depth, height or spacial position (x, y, z). As the University doesn't have a robot yet, those pictures will be provided by the University.

After having those input images, the robot will have to :

1. process the input images of his surrounding, in order to create a DEM
2. generate the 3D map by analysing the DEM
3. analyse this 3D map and take a decision (knowing his previous decisions, if they exist), it will decide were to go
4. analyse his decision, and learn from it



**Figure 1:** General functioning

The following sections will present the DEM generation and the technics we will use in order to perform it.

### 3.1 DEM generation

The first part of the process will be handling the input images and apply a STRT on them. So the STRT would work, it is important to have  $2^k$  number of pixels in each images. By performing a STRT and a Inverse Set Theoric Rajan Transform (ISTRT) on one image, we will get a filtered image, with less noise.

After the noise has been removed from all of the image, we will detect eadges on it by transformng the image into a monochrome image (with only 2 values, 1 and 0) and apply a eadge detection algorithn.

By detecting the eadges, we will be able to identify each objects on the image, and extract those objects.

### 3.2 Rajan Transformation and it's application in the sets

In this section, we will present the forward rajan transform and it's application in the set domain. The reverse transforms won't be presented. Every equations, figures and explanations have been inspired by the following reference RAJAN, 2006.

#### 3.2.1 Rajan Transformation

The rajan transform take a sequence of  $2^i$  numbers (the number of element of the sequence have to be a power of 2), transform it, and return another sequence of  $2^k$  numbers. We will call  $x(k)$  the input sequence,  $X(k)$  the output sequence of the tranform, and  $N$  the number of element in the sequence RAJAN, 2006.

$$x = x(0), x(2), \dots, x(k-1) \text{ where } k \text{ is a power of } 2 \quad (1)$$

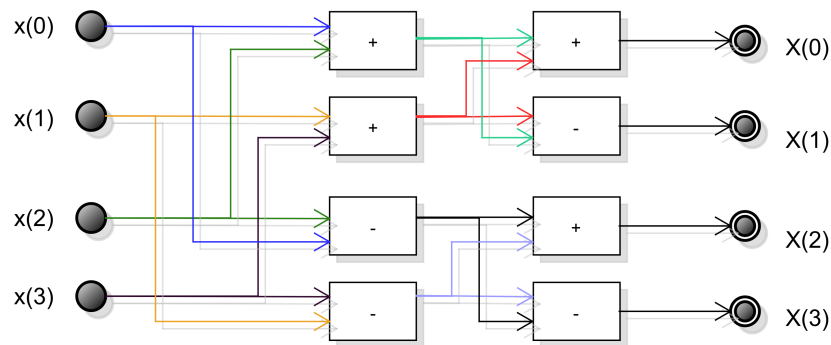
We will define two other sequences  $g(k)$  and  $h(k)$  as following :

$$g(k) = x(k) + x(k + \frac{N}{2}) \text{ with } 0 \leq k \leq \frac{N}{2} \quad (2)$$

$$h(k) = |x(k) - x(k - \frac{N}{2})| \text{ with } \frac{N}{2} \leq k \leq N \quad (3)$$

In other words, the sequence  $x$  will be divided in two. Then we will sum the two first value of the subsequences (which will give us the results of  $g(1)$ ), then the two seconds values (which is  $g(2)$ ), then the thirds values, and so on until all the elements were processed. Then we will do the same operations but with a subtraction instead (to get the sequence  $h$ ).

This processus will then have to be repeated on the subsequences  $g$  and  $h$  separately. We can see here the recursive character of this transformation. The figure 2 illustrate the rajan tranfform in a more procedural way, with a sequence of 4 elements.



**Figure 2:** The forward Rajan Tranfform (RT)

### 3.2.2 Set Theoretic Rajan Transformation

The set theoretic rajan transform is the application of the rajan transform in the set domain, and so instead of transforming a sequence of numbers, it will tranform a sequence of sets, and return another sequence of sets.

In the set domain, the addition correspond to the union, and the subtraction to the difference of two sets.

## 4 Methodology

I am planning to divide this work into 3 parts, which will concern :

1. the DEM generation (which involves Symbolic Computing, and STRT)
2. the 3D Map Generation
3. the Autonomous Navigation and the Reinforcement Learning

During each part, I will document myself on the subject, then write an algorithm, implement it, and finally document my work.

## References

- RAJAN, E. G. (2006, June). Rajan transform and its uses in pattern recognition. Retrieved from <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.180.8418&rep=rep1&type=pdf>
- KOBER, J., BAGNELL, J. A., & PETERS, J. (2013, July). Reinforcement learning in robotics : a survey. Retrieved from [http://www.ias.tu-darmstadt.de/uploads/Publications/Kober\\_IJRR\\_2013.pdf](http://www.ias.tu-darmstadt.de/uploads/Publications/Kober_IJRR_2013.pdf)
- Dogra, A. & Patterh, D. M. S. (2013, August). Performance comparison of gaussian and butterworth high pass filters.
- RAJAN, E. G. (2003). Symbolic computing, signal and image processing.

## Glossary

### C

**CLAP** Cellular Logic Array Processing. 1

**D**

**DEM** Digital Elevation Model. 1, 2

**G**

**GUI** Graphic User Interface. 1

**I**

**ISTRT** Inverse Set Theoric Rajan Transform. 2

**S**

**STRT** Set Theoretic Rajan Transform. 1, 2