Chapter 1

Introduction

All living beings, including humans, need to travel to eat, sleep and breed. To properly move into a new ground its traversability must be estimated. Thanks to millennia of evolutions this task has been effectively solved by mother nature. Every species has learned to use their local sensors to effectively navigate new environments. Humans, for instance, are equipped with a powerful traversability estimator, called vision, that is able to classify grounds by extract features from a local region, a patch, such as elevation, size, and steepness and combined them with the brain to produce a local planner able to find a suboptimal path to cross.

Similarly, ground robots need to walk in order to fulfill their tasks. Taking inspiration from nature, robots must be able to correctly aggregate terrain features to predict their traversability.Two main approaches have emerged over the years to collect the data, geometric and appearance. The first one utilize purely geometric features to label a patch, while the second relies mostly on camera's images. In both cases, those samples are used in supervised learning to train a traversability estimator. While appearance data must be collected directly from a real or simulated environment, geometric features do not depend on the robot's interactions with the ground.

In most indoor scenarios, specific hardware such as infrared, cameras or LIDAR sensors is used to collect real world data during a robot exploration of the enviroment. Even if an agent may be trained only by samples collected in one location, the learned mapping will be generic since most indoor scenarios share similar features across different places shifting the problem from traversability estimation to obstacle avoidance. For example, the floor is always flat without any bumps or holes due to is artificial design. For this reason, most of the times, the classifier learn to traverse a patch by looking if there are obstacles ahead.

On the other hand, outdoor grounds may have less artificial obstacle but they have a not homogeneous ground making challenging to estimate where the robot can properly travel. Also, a given patch with a specific shape may not be traversable by all direction due to the not uniform features of the terrain. On top of that, using real world data may be unfeasible due to the time required to move the robot on different grounds and the possibily to introduce bias if the data samples are not varied enough. Fortunately, ground can be generated articially and used in a simulated enviroment to let the robot walk in it while storing its interactions. Moreover, grounds maps can be easily obtained nowadays by using third-party services such as google maps or flying drone equiped with mapping technologies.

We proposed a full pipeline to estimate traversability tested on legged crocodile-like robot on uneven terrain based entirely on data gather trought simulation. We first generated thirty synthetic maps encoded as heghtmaps with different features: bumps, walls, slopes, steps and holes. Then, we run the robot on each one of them for a certain amount of time while storing its interactions with the enviroment, position and orientation. Later, the procede to create the dataset by cropping a patch for each stored data point in a such way that includes the robot's footprint and the maximum possible amount of ground that can traverse in a selected time window. After, we label each patch by traversable or not traversable if the robot's advancement computed in that time using the same time window is less or greater than a treshold. The treshold depends on the robot's locomotion and is calculated by spawing it in front of different obstacle and observing its advancement, we set the value for the tested robot to twenty centiments. Those patches are used to fit a deep convolutional neural network to prediction the traversability leaving to the network the task to extract important ground's features.

The report is organized as follow, the next chapter introduces the related work, Chapter 2 describes our approach, Chapter 3 talk in deep about the implementation details, Chapter 4 shows the results and Chapter 5 discuss clusion and future work