Universidade de Aveiro Departamento de Engenharia Mecânica
2014

Ricardo Luís da Mota Silva

Unidade Amovível de Hodometria para Veículos com Direção Ackermann

Removable Odometry Unit for Vehicles with Ackerman Steering

Universidade de Aveiro Departamento de Engenharia Mecânica
2014

Ricardo Luís da Mota Silva

Unidade Amovível de Hodometria para Veículos com Direção Ackermann

Removable Odometry Unit for Vehicles with Ackerman Steering

Dissertação apresentada à Universidade de Aveiro para cumprimento dos requisitos necessários à obtençãoo do grau de Mestrado em Engenharia Mecânica, realizada sob orientação científica de Vítor Manuel Ferreira dos Santos, Professor Associado do Departamento de Engenharia Mecânica da Universidade de Aveiro e de Ricardo José da Silva Pascoal, Estagiário Pos-doc no Departamento de Engenharia Mecânica da Universidade de Aveiro.

O júri/ The jury

Presidente / President

Prof. Doutor Jorge Augusto Fernandes Ferreira
Professor Auxiliar da Universidade de Aveiro

Vogais / Committee

Prof. Doutor Vítor Manuel Ferreira dos Santos
Professor Associado da Universidade de Aveiro (orientador)

Doutor Procópio Silveira Stein
Investigador, Inria Grenoble Rhône - Alpes

Agradecimentos / Acknowledgements

Ao Professor Doutor Vítor Santos, pelo acompanhamento, por todo o entusiasmo e apoio prestado. O seu espírito crítico e ideias foram peças fundamentais para este projeto.

Ao Doutor Ricardo Pascoal, por estar sempre disponível a dispender o seu tempo. A sua ajuda foi preciosa.

Um agradecimento especial à Motofil e à Infaimon pelo equipamento fornecido.

Aos meus companheiros de laboratório, por fazerem todos os dias menos difíceis com a boa disposição, companheirismo e inspiração.

À minha família pelo apoio incondicional.

A ti Letícia por estares sempre ao meu lado e por todo o apoio e coragem que me dás.

Palavras-chave

Automóvel, Hodometria, Hodometria Visual, Navegação de Robôs, Robôs Autónomos

Resumo

O principal objetivo deste trabalho é o desenvolvimento de uma solução de hodometria para veículos com direção Ackermann. A solução tinha que ser portátil, flexível e fácil de montar. Após o estudo do estado da arte e uma pesquisa de soluções, a solução escolhida foi baseada em hodometria visual. Os passos seguintes do trabalho foram estudar a viabilidade de utilizar câmaras lineares para hodometria visual. O sensor de imagem foi usado para calcular a velocidade longitudinal; e a orientação da movimento foi calculado usando dois giroscópios. Para testar o método, várias experiências foram feitas; as experiências ocorreram indoor, sob condições controladas. Foi testada a capacidade de medir a velocidade em movimentos de linha reta, movimentos diagonais, movimentos circulares e movimentos com variação da distância ao solo. Os dados foram processados usando algoritmos de correlação e os foram resultados documentados. Com base nos resultados, é seguro concluir que hodometria com câmaras lineares auxiliado por sensores inerciais tem um potencial de aplicabilidade no mundo real.

Keywords Automotive, Odometry, Visual Odometry, Robot Navigation, Autonomous Robots

Abstract The main objective of this work is to develop a solution of odometry for vehicles with Ackermann steering. The solution had to be portable, flexible and easy to mount. After the study of the state of the art and a survey of solutions, the solution chosen was based on visual odometry. The following steps of the work were to study the feasibility to use line scan image sensors for visual odometry. The image sensor was used to compute the longitudinal velocity; and the orientation of motion was computed using two gyroscopes. To test the method, several experiments were made; the experiments took place indoor, under controlled conditions. It was tested the ability to measure velocity on straight line movements, diagonal movements, circular movements and movements with a changing distance from the ground. The data was processed with correlation algorithms and the results were documented. Based on the results it is safe to conclude that odometry with line scan sensors aided by inertial sensors has a potential for a real world applicability.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Project Context

This work arose in the context of the ATLAS project of the Department of Mechanical Engineering of the University of Aveiro, which is inserted in the field of advanced sensing and active systems designed for automobiles and other mobile platforms. ATLAS is a project that was started in 2003 to participate in Robotics Competitions. Since then three scale models were developed and two of them participated in several robotics competitions and won several prizes. The two winning robots are the Atlas20xx and the AtlasMV (figure 1.1). After the success of the project, a full sized prototype was created. This

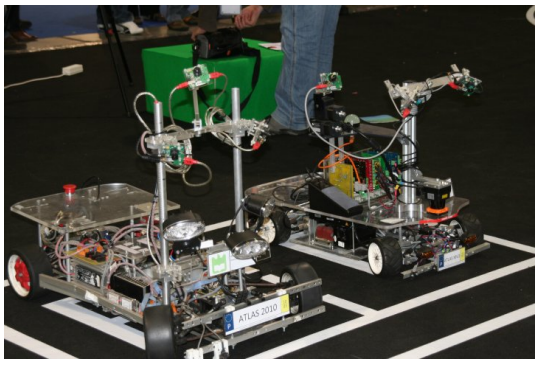


Figure 1.1: The scale models, Atlas20xx on the left and AtlasMV [1].

prototype is the ATLASCAR and its mission is to serve as platform for research on Advanced Driver's Assistance Systems. The vehicle is equipped with several state of the art sensors (figure 1.2). Lasers, Stereo Cameras, IMU, GPS, etc [2]. The main objective of the project is to test algorithms and hardware for driver assistance and safety. The project is evolving to deal with real road scenarios and the mission it to perceive not only what is happening in the environment, but also within the vehicle.

1.2 Motivation

To allow for autonomous driving, driving assistance and monitoring of risk and manoeuvres aboard a car, it is necessary that the computerized system have access to good

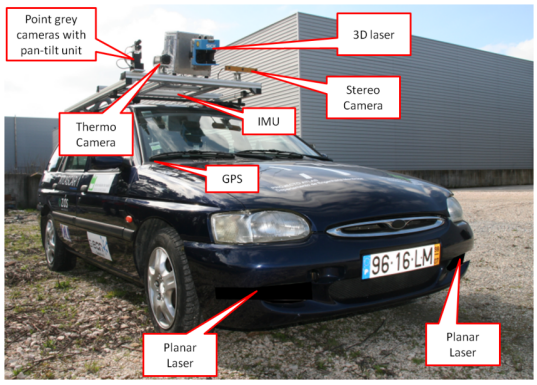


Figure 1.2: The current set of sensors [1].

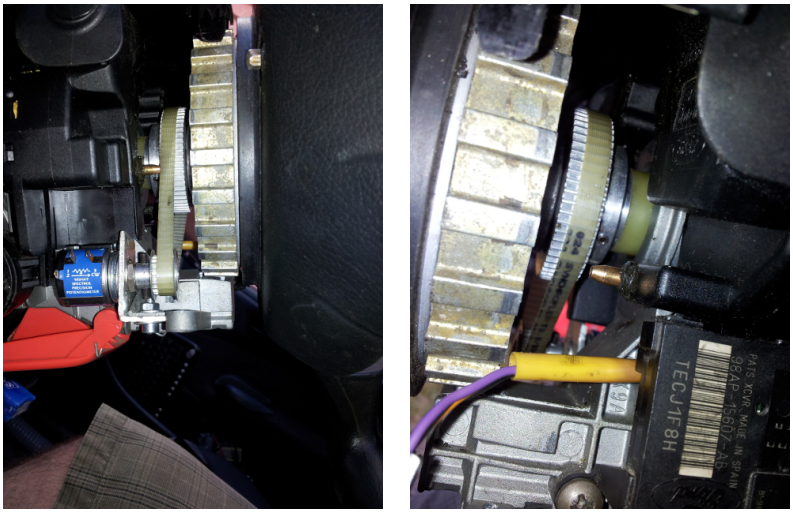
estimates of the vehicle movements, in particular the displacement on the road. Some modern upscale vehicles have this information, but the vast majority of the fleet does not. Thus, arises the need to develop a system to estimate the position of the vehicle wheels using odometry and Ackermann steering constraint. This information along with other informations from the vehicle and the driver are important to active and passive systems for driving assistance and safety.

ATLASCAR already has its own odometry system, the velocity is calculated with an incremental encoder installed on one of the back wheels, the encoder is enclosed in an apparatus fixed in the body of the car (Figure 1.3). The encoder is connected directly to the wheel, so do not have any kind of multiplication. The encoder resolution is 50 PPR this means that with the wheel's diameter of 0.55 meters we can get a new value of the car's velocity every 3.49 centimetres.

Figure 1.3: Velocity measurement system on ATLASCAR [3].

The angle of the steering wheel is measured with a potentiometer connected through a pulley system directly in the steering column. This mechanism is invasive and forced modifications on the structure and devices of the steering column (Figure 1.4). The potentiometer has a 10 $k\Omega$ resistance and the gear ratio through the pulley mechanism is

1:3, the data is collected by a PIC18F258 micro controller with 10-Bit analog-to-digital converter.



(a) (b)

Figure 1.4: Steering wheel angle measuring system [3].

1.3 Objectives

The main objective of this project is to develop an odometry system which is easily installable and non invasive to the vehicle. It should be able to be used on most of the road cars. The system should provide good estimates of velocity, distance traveled and orientation of the car.

1.4 Structure of the Dissertation

The stages of this project are presented in the following chapters. The stages are organized as follows:

- Chapter 2 - State of the art

in this chapter is presented a brief history of the odometry, a study of the odometry solutions and sensors, and finally a survey of the commercial solutions;

- Chapter 3 - Study of the Solution

in this chapter the proposed solution is presented and a study of the algorithms to be used is made;

- Chapter 4 - Experimental Procedure

in this chapter there is made a description of the materials and the experimental procedure used, as well as the calibration of the equipments;

- Chapter 5 - Results and Discussion

in this chapter the experiment results are presented and described as well as its discussion;

- Chapter 6 - Conclusions and Future Work

in this chapter the conclusions and an analysis of the achieved objectives is performed. As well as possible future work will be discussed.

Chapter 2

State of the Art

One of the most important pieces of information for a robot is its position in the environment. In order for an autonomous robot to perform its tasks, its position and orientation must be known. For an mobile robots its position can be classified in two categories: relative positioning and absolute positioning. A relative positioning system is one which uses sensors on the robot's wheels or other navigation systems to calculate the position without taking a fixed consideration. An absolute positioning system is one which uses fixed a reference for position determination, such as the Global Positioning Systems (GPS). One of the most fundamental techniques to position determination is the use of encoders on the wheels, which is called Odometry. Odometry is very important to have precise estimates on the wheel's velocity and orientation and combined with other sensors it makes a reliable way to position estimate [4].

2.1 History of Odometry

Since the early times men found the need to measure distance and velocity. An odometer is the device that measures the distance travelled by a vehicle and it can be electronic, mechanical or the combination of two.

Possibly the first odometer was described by Vitruvius around 27 and 23 BC. The odometer of Vitruvius was based on wheels of 1.2 m diameter turning 400 times, that making one Roman mile, about 1400 m. At each revolution a pin on the axle hit a 400 tooth gear making one revolution per mile. This gear engaged another device that dropped pebbles into a box and thus the distance travelled would be calculated by counting the pebbles. It is unclear if the instrument was ever built at the time. Leonardo da Vinci also drawn an odometer (Figure 2.1) based on the plans [5].

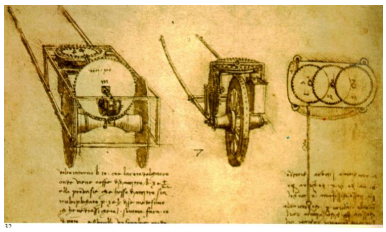


Figure 2.1: Odometer drawn by Leonardo da Vinci.

Blaise Pascal and Wilhrlm Schickard, both, made the invention of the mechanical calculator in the early of the 17th century. Pascal's invention, called Pascaline (Figure 2.2), was made in 1642. Thought it was not a odometer, the Pascaline used gears to calculate measurements. Each gear contained ten teeth, the first gear moved the next gear one position at each complete revolution, this is the principle used in the modern odometers [6].

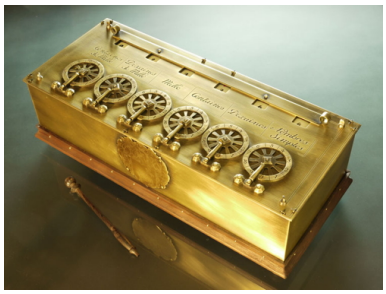


Figure 2.2: Pascaline, Pascal's calculator [6].

In 1698 Thomas Savery, an English military engineer and inventor, developed odometers to measure the distance travelled by ships. In 1775 Benjamin Franklin, a statesman, writer and inventor, built a odometer to measure the mileage of postal routes to analyse the best routes for delivering the mail. In 1847 William Clayton, Orson Pratt, Appleton Milo Harmon, invented the Roadometer that recorded the distance travelled by wagons. The Roadometer was a pascaline inspired device that used gears to measure the distance travelled (Figure 2.3) [7].

Figure 2.3: Roadometer, first used in May, 1847 [7].

Arthur P. and Charles H. Warner produced a device to measure speed of industrial equipment which used a magnet in a rotating shaft to induce a magnetic pull to a metal disk. This device became the first automobile speedometer in 1912 and became standard equipment on most American-made cars [8].

2.2 Odometry

Odometry is the method that uses data from moving sensors to estimate the change in position over time. Odometry is used to estimate, and not determine, the position of a vehicle relative to a starting location. This method is sensitive to errors due to the integration of velocity measurements over time to give position estimates. A robot can have, for example, rotary encoders on its wheels or on its legged joints and use the data from this sensors to measure how much the wheels have rotated, and knowing the circumference of the wheel it can calculate the distance travelled.

Odometry mostly uses encoders to measure wheel rotation and/or steering orientation, resolvers also can be used but encoders are usually the first choice. Odometry has the advantage that it can be self-contained, and it is always capable of providing an estimate of position of a vehicle. The disadvantage of odometry is that it must integrate displacement measurements to give position estimates, this causes this method to be sensitive to errors. The position error grows over time and an independent reference should be used periodically to reduce the error [9].

Optical encoders are the most common sensors used in odometry. The main components of the optical encoder's is a disc made of glass or plastic with transparent and opaque areas, a light source and photo detector array that reads the pulses generated by the optical pattern from the disc's position. Increasing the number of pulses increases the resolution. These type of devices are relatively inexpensive and well suited for velocity feedback low to high speed systems. Some encoders have two channels displaced one from the other and by determining which one is the leading channel it is possible to calculate the direction of rotation. The addition of a channel also has the benefit of increasing the resolution (Figure 2.4). There are some downsides associated to this sensors, in the event of a power interruption all relative position information is lost, this sensors are also more sensitive to damage by external agents [10].

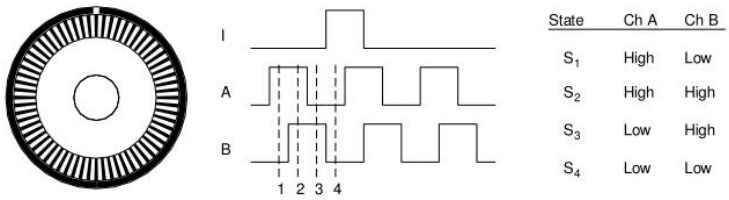


Figure 2.4: The phase relationship between channel A and B can be used to determine the direction of rotation. With the unique states S1 to S4 it is possible to increase the resolution with a multiplication factor of 4 [11].

Another type of optical encoder which can be used to measure wheel's orientation is the absolute encoder. The application of this type of sensor is mostly associated to slower rotational applications where the loss of the position information cannot be tolerated. Absolute encoders produce a unique digital code for each distinct angle of the shaft (Figure 2.5). Each track of the disk codes a bit, increasing the number of tracks increases the resolution and also increases the diameter of the disk and consequently the decrease in shock and vibration tolerance. Absolute encoders are best suited for slow rotating systems such as direction angle. The main downside of this type of sensor is the increasing fragility and cost as the resolution increases [10].

Encoders are the most commonly used sensors to odometry purposes. They provide a straightforward encoding scheme and digital output, this results in a low-cost reliable package with good noise immunity [10].

Encoders like the ones used on the ATLASCAR are inexpensive and higher resolution encoders are highly available and still at a low cost. 1024 pulse encoders are very common and installed on the car would provide a speed measurement at every 1.7 millimetres, which is a very high resolution. This method still has the disadvantage of having an error accumulation that grows over time. Wheel's geometry changes are the most common

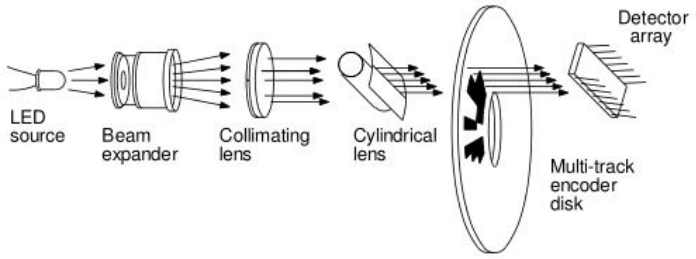


Figure 2.5: The line of light passes through the coded pattern of the rotor and that corresponds to a unique code that specifies the absolute angular position [12].

source of error due to fluctuation of tire pressure, tire diameter, slippage of the wheels and tire tread wear.

Most of the modern cars, the ones with ABS systems, use a speed sensor based on inductive principles. Coupled to the wheel's axle there is a spur gear and next to it there is an inductive sensor that generates a pulse every time a tooth passes (Figure 2.6). This kind of sensors is very robust, they do not need contact between the moving parts and the sensor is immune to water and dust. Although, they have some disadvantages, measurements may be affected by magnetic fields, also they do not provide accurate measurements for low speed movements [13].

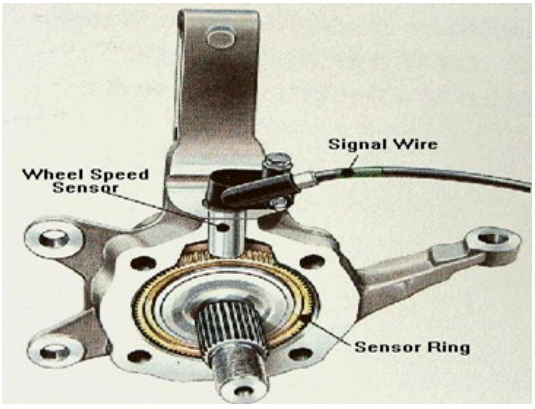


Figure 2.6: ABS inductive velocity sensor.

2.3 Inertial Navigation

This method measures rate of rotation and acceleration with gyroscopes and/or accelerometers. The measurements of a gyroscope must be integrated once to get angular displacement and accelerometer data must be integrated twice to obtain linear displacement. This sensors have the advantage that they are self-contained and they the measurements are independent of the wheel's geometry changes and wheel's slippage. On the other hand this type of inertial devices have a natural drift which is a source of error, this drift accumulates over time causing a deviation on the measurements. This data then needs to be integrated to obtain velocity and position and the error grows without

bounds. For this reason, inertial sensors are unsuitable for accurate positioning over an extended period of time [10].

Along with odometry, gyroscopes are the most used sensors for determining the orientation of a mobile robot. Gyroscopes can be classified in several categories, the most common types of gyroscopes are mechanical, optical and piezoelectric vibrating gyroscopes.

The mechanical gyroscope is a rotation sensor based on the inertial properties of a rapidly spinning rotor, this technology is known since the early 1800s. This gyroscope is composed by a rotating mass mounted on a support with three moving axes (Figure 2.7). The mass is rotating at a constant velocity and when the support is inclined causes gyroscopic pressetion, which is a rotation of the support to compensate the inclination. A modern mechanical gyroscope provides a voltage or frequency output signal proportional to the turning rate. High precision mechanical gyroscopes are expensive, also they are big on size and heavy. For this reasons mechanical gyroscopes have a limited application in mobile robotics [10].

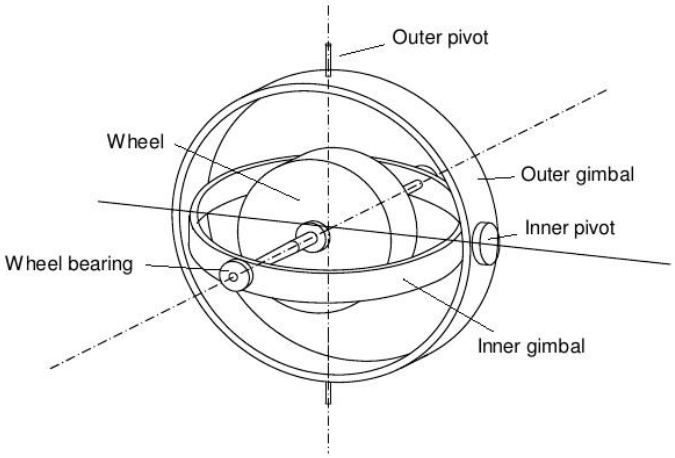
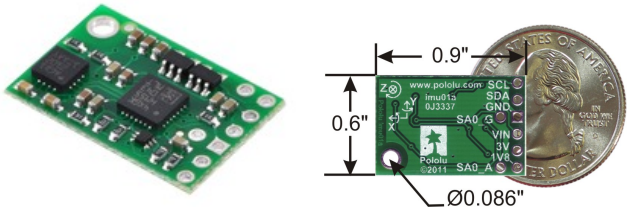


Figure 2.7: Typical mechanical gyroscope configuration [10].

Piezoelectric vibrating gyroscopes use Coriolis forces to measure rate of rotation. In a typical configuration three piezoelectric transducers are mounted on the sides of a triangular prism. If one of the transducers is at a constant resonance frequency the vibrations will be transmitted at equal intensity to the others transducers. If a rotation is applied to the system the other two transducers will measure vibrations with a difference. This voltage difference is proportional to the rate of rotation [10]. Inexpensive vibrating structure gyroscopes using MEMS technology, a technology for microelectronics, have become widely available. These are packaged into integrated circuits providing analog or digital outputs or sometimes both. Some units integrate gyroscopes for multiple axes in a single part. Some units even incorporate multiple-axis gyroscopes and accelerometers to provide a six degrees of freedom output. These units are called Inertial measurement units, or IMUs (Figure 2.8). This type of sensor is the most used in consumer electronics like smart-phones due to their small size and low energy consumption. They are also used in the automotive and industrial robotic field. Some units became so inexpensive that they are also used by hobbyists [14].

The other type of gyroscope is the optical gyroscope. Optical gyroscopes are build



(a) (b)

Figure 2.8: POLOLU - MinIMU9DOF, an inertial measurement unit (IMU) that has a 3-axis gyro, a 3-axis accelerometer and a 3-axis magnetometer [15].

with no moving parts, being a maintenance free device. Its design is based on two fibber optics coils with the same length on which two laser beams with opposite directions travel. If a rotation is applied the duration of the travel by each laser beam will be different, this is known as the Sagnac effect. Recently optical gyroscopes have become available at competitive prices being the choice for applications with high precision needs [10].

2.4 Visual Odometry

Visual odometry is a technique that estimate robot location using visual information (images); the position is defined as the incremental estimation of robot motion from image sequences using an on-board camera. Visual odometry is a method highly used on off-road robots like space robotic exploration missions. The motion of the robot is estimated with calculation of the pixel displacement between two consecutive frames, this is called optical flow [16].

The optical flow is mostly calculated based on features that stand out in the image, this method doesn't work well on smooth or texture-less surfaces like concrete and asphalt roads. The other method, with better results, is template matching. This method use a small area of a image and tries to find it in the next frame, if found, it is possible to calculate the optical flow [17].

There are two main different approaches in hardware, the first uses a camera looking forward and the velocity is calculated based on the movement of the surroundings. This method is very sensitive to lighting changes and a high quality image is needed. The other method uses a camera looking directly at the ground, this is the principle used in most optical mice. This method needs simpler computation and hardware and lighting conditions can be handled better [18].

The literature suggests visual odometry as a better odometry system in terms of precision and error accumulation. Navid Nourani-Vatani et al [17, 19], obtained good results using a common webcam at 20 fps and a image resolution of 640 x 480 pixels; they used template matching with a searching area of 320 x 320 pixels. The acquisition speed was set to 20 fps due to the high processing time, 42 ms/frame, this is a clear limitation. Also due to the low frame rate, the velocity of the vehicle was limited to 1.5 m/s, which is very low. Nevertheless, this method showed better results than common wheel's odometry. This method is illustrated in Figure 2.9; the image displacements are translated to vehicle displacement to estimate its position and trajectory.

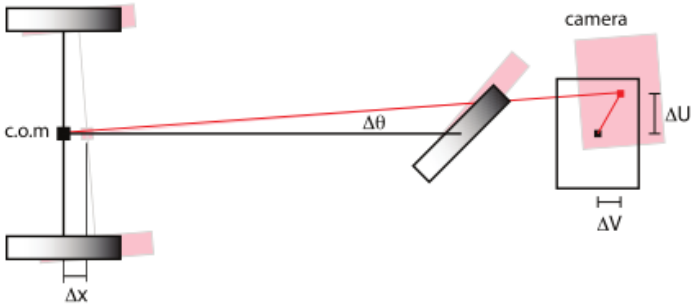


Figure 2.9: Simplified model of a vehicle with Ackermann-like steering model. The pixel displacement, ΔU and ΔV, translates to vehicle displacement Δx and $\Delta \theta$, which are defined with respect to the center of motion corresponding to the midpoint of the rear axle [17].

The velocity allowed by the measuring unit is are mostly influenced by the sampling rate and the size of the image. If the velocity is higher than the limit, there will be no overlap of the frames and there will be no similarity between the successive images; this will cause the impossibility of the estimation of velocity [18]. So, for car-like velocities it is necessary a very high sampling rate camera, this brings difficulties as the computational power needs to be considerable. Viktor Kálmán [18] suggests the use of line scan cameras to achieve high velocities and resolution, using less computational resources than a common matrix camera. In his work he made simulations and came to the conclusion that it is possible to measure movements of vehicles up to the range of 100m/s with high accuracy.

Visual odometry has clear advantages over the common wheel's odometry. It is possible to measure velocity independent of the variations in tire pressure, tire diameter, uneven terrain and tire slip. It is possible to obtain a compact and easy to install module. However, there are several drawbacks, the lighting is an important factor (and it isn't always possible to control), also the sensors are high price and sophisticated.

2.5 Commercial Solutions

Every car on the roads have a odometer on the dashboard that shows the instantaneous velocity, although most of them have a odometer actuated mechanically so that there is no electrical signal that can be measured.

There are already some companies that sell devices capable of measuring wheel's speed and orientation. Most of this devices are used by automotive manufacturers to test the dynamics and durability of their products.

The company Kistler Automotive sells various sensors that can be used to measure velocity and direction of a car, they provide one product that measures wheel's rotation and others that are optical and measure velocity and orientation without contact. The WPT Sensors is a sensor that is universally adaptable for acquisition of vehicle wheel speed (Figure 2.10). This sensor consists of an optical incremental encoder and has 1000 pulses in its standard configuration. Its applications are wheel slip measurement,

acceleration and braking tests, ABS testing. [20]

Figure 2.10: WPT sensors from Kistler Automotive [20].

Kistler has another kind of solution based on optical sensors, the 2-Axis Non Contact Optical Velocity and Slip Angle Sensors. These sensors provide slip-free measurement of distance, velocity and slip-angle which is the angle between a rolling wheel's actual direction of travel and the direction towards which it is pointing (Figure 2.11).

Figure 2.11: 2-Axis Non Contact Optical Velocity and Slip Angle Sensors [21].

Figure 2.12: POS LV

Applanix provides a solution based on GPS and inertial sensors, the POS LV (Figure 2.12). POS LV is a position and orientation system, utilizing integrated inertial technology to generate stable, reliable and repeatable positioning solutions for land-based vehicle applications.The product generates a true representation of vehicle motion in all three axes, works in areas of intermittent, or no GPS reception, computes wheel rotation information to aid vehicle positioning, embedded GPS receivers provide heading aiding to supplement the inertial data. [22]

Chapter 3

Study of the Solution

The main objective of this project is to develop an odometry system which is easily installable and non invasive to the vehicle. The system should provide good estimates of velocity, distance traveled and orientation of the car. This chapter contains a study of the solutions including a final proposed solution.

As stated in the State of the Art Chapter, there are three kinds of odometry solutions to solve this problem, common odometry with wheel's sensors, visual odometry and inertial sensors.

Visual odometry and inertial sensors have great advantages over the common odometry, mostly because this sensors are small, compact and can be mounted in a non-invasive way. When using wheels sensors there is the need to develop an apparatus to support them and this apparatus needs to be easily installable.

Using only inertial sensors is not good, as to obtain the displacement one needs to integrate the acceleration, from accelerometers, two times and to obtain orientation we need to integrate angular velocity, from gyroscopes, one time. Integrating acceleration introduces cumulative errors due to noise which makes the inertial sensors the least suitable for the objective.

Actual GPS equipments became affordable and accurate enough for speed measurement. However they depend on satellite visibility, which makes it ineffective in environments with obstacles and indoor locations.

In the following subsections two of the possible solutions will be presented, a mechanical solution using encoders to measure wheel's velocity and orientation; and a visual odometry solution using a line scan camera to compute velocity plus a gyroscope to compute orientation.

3.1 Mechanical Solution

For a solution using encoders there are several ways and combinations to solve the problem. The next topics are the chosen solutions to be discussed.

- One encoder in each of the rear wheels
  - with this configuration it is possible to calculate the velocity and orientation with the difference in velocity on the two wheels. This is a simple configuration, with low setup complexity and low cost. Although we can calculate the

vehicle orientation angle, this system doesn't provide the front wheel orientation which is a very important information;

- Encoder on one of the wheels and angle monitoring of the steering wheel
  - This solution implies the existence of some device on the inside of the vehicle cabin; it is very hard to install a device on the steering wheel without it being an obstacle to the driver;
- Encoder and angle monitoring in a single front wheel
  - This is a practical solution because with one single device we can monitor all the wanted magnitudes, this makes it a good solution in terms of compactness and easy to setup. The major downside is the complexity of the apparatus needed to support the sensors to keep with all the front wheel degrees of freedom;
- Encoder in one of the rear wheels and angle monitoring of one of the front wheels
  - With this configuration we can monitor both wheel's velocity and steering angle. There is the downside of the need of two separate devices, one for the front and one for the rear wheel. However, this is a low complexity system to produce and setup.

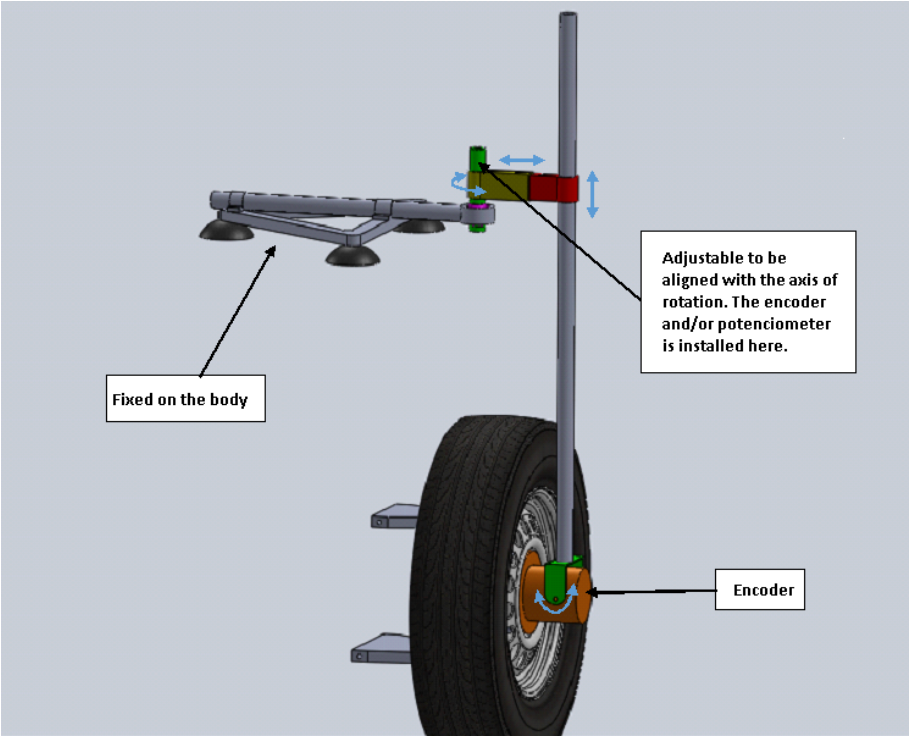


Figure 3.1: Illustration of a possible mechanical solution and its main components.

A mechanical solution is proposed, as in the Figure 3.1. The solution consists of an apparatus mounted on one of the front wheels to serve as a support for an encoder for

wheel's velocity measurement and another encoder or potentiometer for angle measurement. As shown in Figure 3.2, the equipment is fixed to the car's body with suction cups and there are several adjustments to meet the geometry variability of the vehicle.

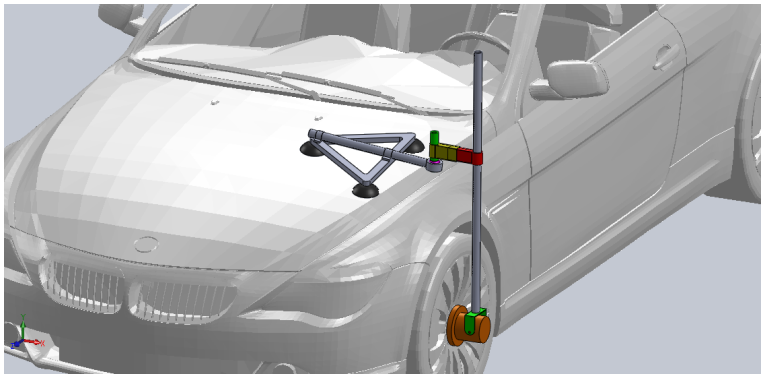


Figure 3.2: Illustration of the device assembly.

This solution provides all the necessary information in only one wheel. A high resolution encoder can be used to provide a high sampling rate of velocity. For the angle of the wheel, a multi-turn potentiometer or an absolute encoder might be used to provide an absolute positioning of the steering angle. The system can be removed and installed on a variety of vehicles due to the adjustments that it provides. However, the rigidity of the system and its constraints can make it fragile for a more dynamic and aggressive driving situation. Also the system being out of the bounds of the car's body may limit its use due to being to exposed to the environment. Given these limitations, the optical solution will be further detailed and studied.

3.2 Optical Solution

Horn et. al. [23] used two cameras on a robot, one looking forward to estimate yaw rate and forward velocity and the other facing down to estimate two dimension velocity. They found that the camera facing down gave the best results for longitudinal and lateral velocity. The tests were made at low speed, lower than 1 m/s.

Nourani-Vatani et. al. [24] also used a similar method with a ground-facing matrix camera mounted on a passenger car. The experiments were at around 1m/s. The velocity measurements had errors comparable to the wheel speed sensor and the cumulative odometry errors were at around 5% after 100m traveled.

It is clear that common matrix cameras are not suitable for odometry to vehicles with car like speeds. Even with high speed matrix cameras, the amount of data to process would be extremely difficult to compute in real-time.

In his PhD thesis, Kálmán [18], suggests the use of line scan cameras to estimate robot's longitudinal velocity. His method suggests that one can compute correlation between successive lines and with that information compute the pixel displacement between them. Line scan cameras provide high resolution and high frame rate at a lower cost than matrix cameras, this makes them suitable to be used at higher speeds. Kálmán created a simulator in which he tested the ability for line sensors to be used to estimate robot's

velocity; he found that a sensor such as the S3901 from Hamamatsu, which works at 15600 fps, can measure velocity to a maximum of 120 m/s or 432 km/h.

Instinctively, one might think that image overlap is impossible when only a line is captured from the ground. Wide pixel sensors or the integrating effect from the lens can be used to capture more information from the ground. Figure 3.3 from [18], shows the integrating effect of the lens on a line sensor. On the left we can see the area being projected on a matrix camera pixels and on the right we can see that the same area is projected on a single line of a line scan camera, providing an integrating effect.

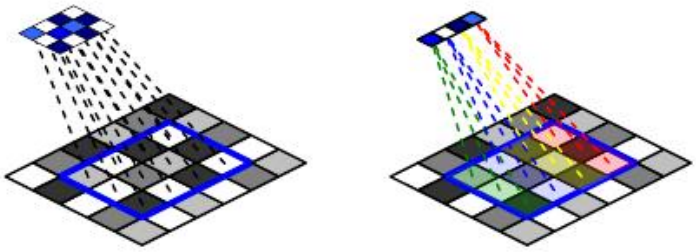


Figure 3.3: Illustration of the imaging of a matrix camera and a line-scan camera with cylindrical lens [18].

A line scan camera can only detect movements that are parallel to its axis, so it only provides one dimensional information. This brings the problem that the sensor needs to ensure good measurements even when movements perpendicular to its axis happen. This happens when the vehicle has a circular or perpendicular movement, this can induce measurement errors as the successive line might have a pattern change from the previous one. This is illustrated in Figure 3.4 from [18]. On the left we can see a parallel movements and it is clear that the successive snapshots can have overlap and a common area in which we can compute correlation. On the right we can see the problem of a sideways movement causing the overlap to be smaller, which will cause a weaker correlation and induce error. This error can be reduced using the integrating effect from the lens, a wider pixel sensor or by a larger field of view. Also the sampling rate can be high enough that the successive snapshots will a small displacement and thus capture the same texture elements.

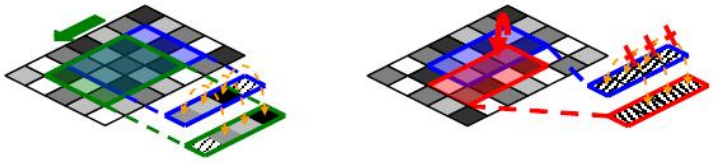


Figure 3.4: Illustration of the problem of sideways motion [18].

3.3 Solution Description

The velocity of vehicles and mobile robots can be measured in many ways. The best solution depends mostly on the application. The type of sensor to be used may also

depend on the platform kinematics.

Optical sensors provide the most information, and with the current processing capabilities available, these sensors are widely used today. The main advantage is that they provide information to movement estimation which is independent to the locomotion system. A measuring system using optical sensors also can be compact and self-contained making it easier to install, this is important as one of the objectives of this work is to develop a system which is removable.

For the reasons stated, the optical solution is the most suited for the objectives of this thesis. The mechanical solution has constraints which may limit its use on some vehicles, has a complicated setup and causes a visual impact. With the given potential of the optical sensors, this work will focus on the study of an optical based solution.

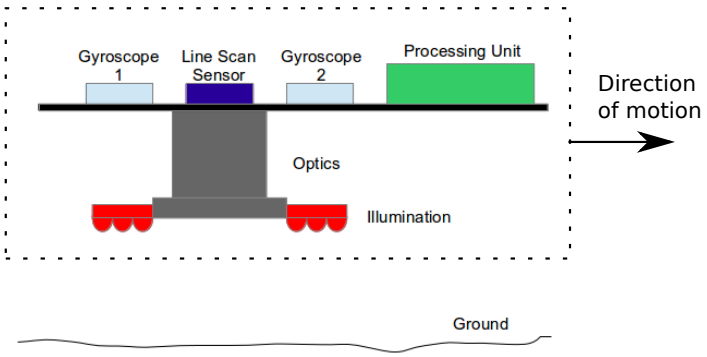


Figure 3.5: Optical solution proposed.

The optical solution proposed is based on a line scan sensor. These sensors are only one dimensional and, thus, only provide longitudinal velocity. There is the need of another sensor to estimate the orientation, to do so, the use of gyroscopes is proposed. On Figure 3.5 there is a schematic of the solution proposed. The main components are as follows:

Line scan sensor: the main component is the line scan sensor; as said before, it should have a high rate and, if possible, have wide pixels to provide an integrating effect and thus capture more information from the ground. Several line sensors are available, the LIS-770i from Panavision Imaging [25] and the S3901 from Hamamatsu [26] both provide wide pixels. The RPLIS2K-EX form Dynamax Imaging [27] is a high resolution sensor (2048x1), they also provide a 2048x4 resolution sensor, the DLIS2K, which can be a interesting solution to the problem of sideways movements.

Gyroscopes: The use of gyroscopes is for orientarion measurements. For more reliable readings the use of two gyroscopes is proposed. As stated by Rafeiro [28], using redundancy from two gyroscopes decreases the errors from the gyroscope's drift.

Optics: this unit should be used to provide the adequate field of view and also the integrating effect. The distance to the ground is constantly changing do to suspension motion and unevenness of the ground, resulting in a variable field of view and miscalibration, which can induce measurement errors. To solve this problem a telecentric lens can be used; these lenses have a magnification which is invariant to the distance [29]. Magnification of a conventional lens can be made invariant to defocus by simply adding an aperture at an analytically derived location [30].

Illumination: the illumination proposed is a high power LED array. Because of the low exposure time, high speed camera sensors need a big amount of light to have a good quality image. Infra-red illumination is the best solution, specially if the sensor is to be used on a car and the light might be a distraction.

Processing Unit: the processing unit needs to be able to do the acquisition of the high speed sensor and process the data. Microprocessors might not be suitable to do that as line scan sensors usually have a pixel clock that works in the MHz range and common microprocessors are not able to do acquisition at that rate. Thus the proposed is a unit composed of a FPGA or CPLD, which are equipments commonly used for high speed data computing and can do parallel computing which speeds the process.

3.4 Principles of Linear Image Sensors

An image sensor is a device that converts an optical image into an electric signal. They are used mostly in digital cameras and other imaging displays. The early sensors were video camera tubes; the modern sensors are semiconductor charge-coupled devices (CCD) or active pixel sensors in complementary metal-oxide-semiconductor (CMOS), Figure 3.6. Both have the same basic principle of converting light to an electric signal [31].

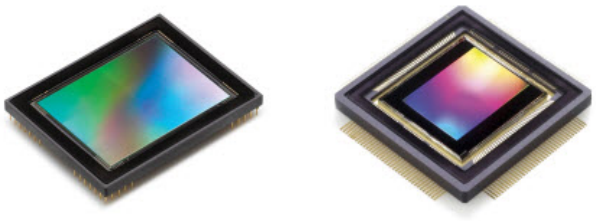


Figure 3.6: Image sensors from Teledyne DALSA - CCD (left) and CMOS (right) [31].

A CCD is a device in which every pixel's charge is converted to voltage through a very limited of outputs, often just one. After the conversion the output is sent as an analog signal. They have the advantage of high uniformity that contributes to image quality. In a CMOS sensor, each pixel has its own hardware for charge to voltage conversion and other circuitry so the chip output is completely digital. This has the disadvantage of increasing complexity and reduced area for light capture. The uniformity of the pixels is lower due to each pixel having its own voltage conversion, but with that it is the advantage of high speed [31].

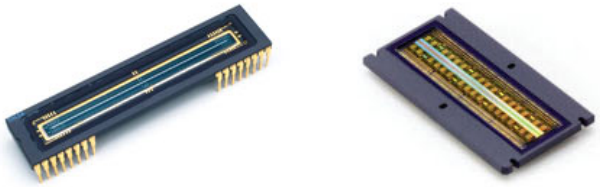


Figure 3.7: Linear image sensors from Teledyne DALSA - CCD (left) and CMOS (right) [31].

The solution proposed in this work is based on line scan image sensors. This sensors are also available as CCD or CMOS, like the ones shown in Figure 3.7. Most of the applications of linear sensors in machine vision require high speed and high frame rates, for this reason CCDs are a technology in decay. On Figure 3.8 it is possible to see how a linear image sensor works. A CCD has usually one charge to voltage converter that operates for all the pixels, opposed to the individual converters on a CMOS. On a CMOS the conversions are made in parallel so the image acquisition is extremely fast.

Line scan cameras usually use multiple outputs methods to increase image sampling rates. Typical line-scan cameras use the following output methods: single tap, dual taps, triple taps, quad taps, and octal taps [32].

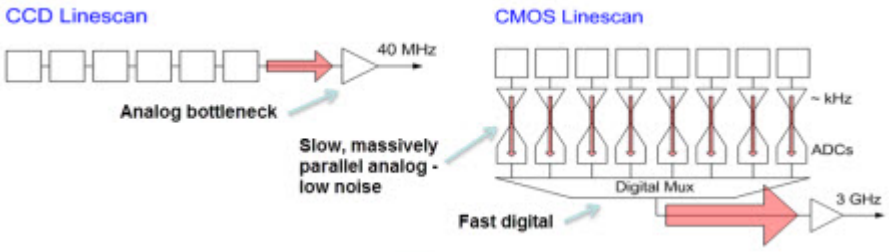


Figure 3.8: Illustration of the working principles of a CCD vs CMOS [31].

The most common output methods in line scan cameras are the single tap, dual taps and quad taps [32]. The single tap is a method in which the photodiodes of a linear image sensor converts the light captured into an electrical signal and transmit it through a single output, this method is illustrated in Figure 3.9.

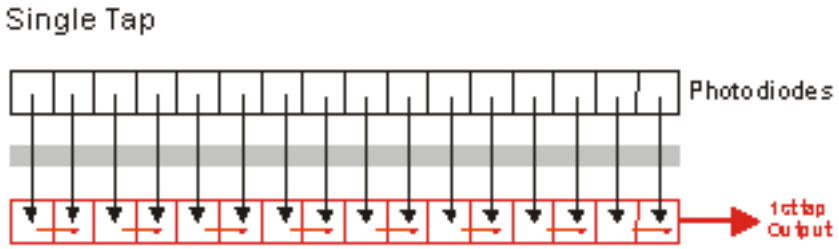


Figure 3.9: Illustration of the single taps data output method [32].

The dual tap is a method in which the photodiodes of a linear image sensor converts the light captured into an electrical signal and transmit it through a dual output. Dual taps can be of two different types, the first is *Even/Odd Output*; in this method separates the light captured into even and odd components to be converted to electrical signal. This method is illustrated in Figure 3.10. The other method for dual tap is the *Front and Rear Output* and the main diference from the previous one is the it separates the linear image sensor in front and rear components, as illustrated in Figure 3.11.

The quad taps are a combination of the two dual taps methods. In this case the two sets of even and odd components are divided each by front and rear to generate four outputs. This process is illustrated in Figure 3.12.

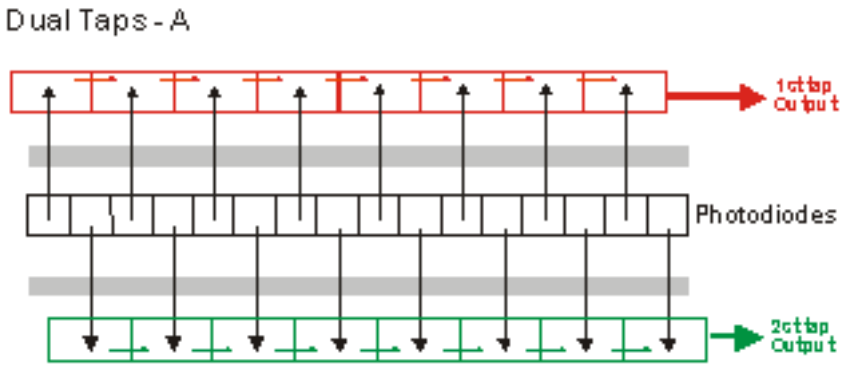


Figure 3.10: Illustration of the dual taps - even/odd data output method [32].

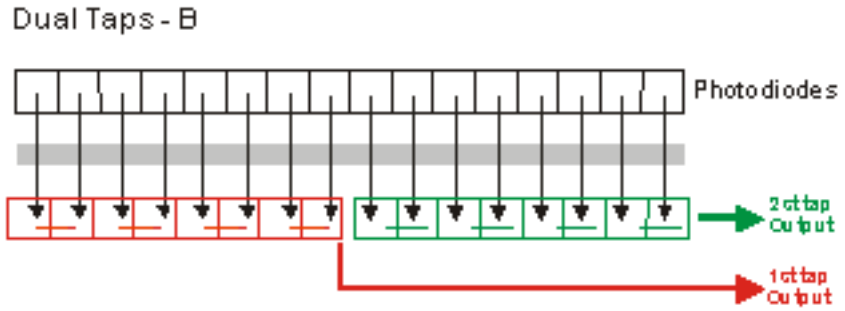


Figure 3.11: Illustration of the dual taps - front/rear data output method. [32].

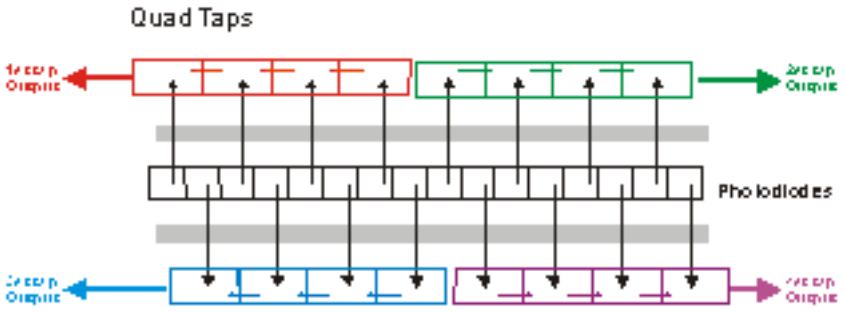


Figure 3.12: Illustration of the quad taps data output method. [32].

3.5 Ackermann Steering Geometry

The purpose of the Ackermann steering geometry is to avoid the necessity for tires to slip sideways when following a curved path. This geometry assures that all the wheels have the axles arranged so that when describing a curve they point to the same center point. As the rear wheels are fixed together, this center point needs to be a line extension of the rear axle. For the axles of the front wheels to intersect this centre point it is required that the inside wheels is turned at a greater angle than the outside one [33].

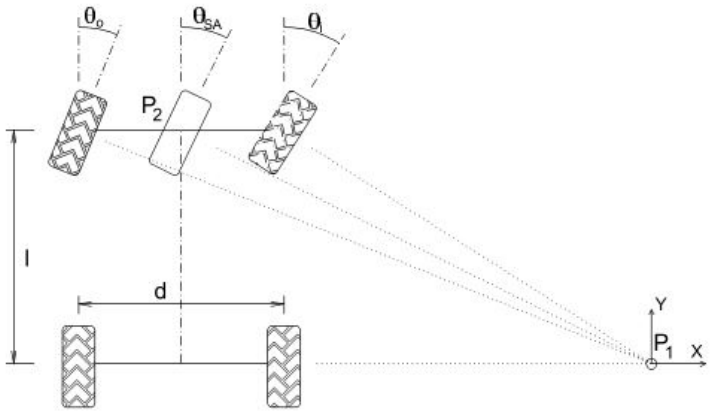


Figure 3.13: In an Ackerman-steered vehicle, the extended axes for all wheels intersect in a common point [10].

The model of the Ackermann steering geometry is illustrated in Figure 3.13. Such a geometry satisfies the Ackermann equations [34]:

$$cot(\theta_i) - cot(\theta_o) = \frac{d}{l} \tag{3.1}$$

where: θ_i is the relative steering angle of the inner wheel; θ_o is the relative steering angle of the outer wheel; l is the longitudinal wheel separation and d is the lateral wheel separation.

The vehicle steering angle θ_{SA} can be simplified as the angle of an imaginary wheel located at the center, P_2 shown in the Figure 3.13. θ_{SA} can be expressed in terms of either the inside or outside steering angles (θ_i or θ_o) as follows [34]:

$$cot(\theta_{SA}) = \frac{d}{2l} + cot(\theta_i) \tag{3.2}$$

or,

$$cot(\theta_{SA}) = cot(\theta_o) - \frac{d}{2l} \tag{3.3}$$

Modern cars now used an Ackermann derived steering and not a *pure* Ackermann. This is due to dynamic concerns where a traditional geometry has disadvantages, but for low-speed maneuvers the principle is the same [34]. Ackermann-like steering provides good traction and ground clearance for all-terrain operation. Ackerman steering is thus the method of choice for outdoor autonomous vehicles.

The characteristics of this steering geometry are important as if the image sensor is mounted on one of the car's axles, the line captured will always be tangential to the arcs described by the wheels. This means that Ackermann steering, under normal conditions, ensures that no sideways movement is made. This assumption is of great importance to the solution proposed as this is essential to obtain correlation between consecutive lines. The solution proposed is not suitable for high lateral displacement vehicles, such as omnidirectional vehicles; for these a different combination of sensors is required.

3.6 Correlation Methods

Two successive images of a line sensor are only two vectors of 1 x n dimensions where n is the resolution of the sensor. This fact makes it possible to use simpler algorithms than the ones needed to process 2-D images from matrix sensors. Correlation between 2-D images is based on optical flow algorithms and template matching techniques. When dealing with 1-D samples there are several methods to compute its similarity. Kálmán [18], suggests the use of similarity coefficients such as Pearson' correlation, cosine similarity and others, described in following subsection. An obvious and effective method to compute similarity between two samples is cross-correlation, widely used in signal processing, described in subsection 3.6.2.

3.6.1 Similarity Coefficients

Euclidean distance - is the distance between two points in a straight line. For the points $P(p_1, p_2, ..., p_n)$ and $Q(q_1, q_2, ..., q_n)$ the distance is calculated as follows:

$$d(p,q) = \sqrt{\sum_{i=1}^{n} (q_i - p_i)^2} \tag{3.4}$$

Manhattan distance - is the distance between two points in a grid-like path, with a strictly horizontal and/or vertical path, as opposed to the diagonal or straight line distance:

$$d(p,q) = \sum_{i=1}^{n} |q_i - p_i| \tag{3.5}$$

Pearson correlation - Pearson correlation coefficient measures the linear correlation between two variables $X(x_1, x_2, ..., x_n)$ and $Y(y_1, y_2, ..., y_n)$. The coefficient can take a value between +1 and -1, where 1 is total positive correlation, 0 is no correlation, and -1 is total negative correlation. For a sample X_i and Y_i. the Pearson correlation coefficient is:

$$r = \frac{1}{n-1} \times \sum_{i=1}^{n} \left(\frac{X_i - \bar{X}}{s_X} \right) \left(\frac{Y_i - \bar{Y}}{s_Y} \right) \tag{3.6}$$

where

$$\frac{X_i - \bar{X}}{s_X} \tag{3.7}$$

$$\bar{X} = \frac{1}{n}\sum_{i=1}^{n} X_i \tag{3.8}$$

$$s_x = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n} \left(X_i - \bar{X}\right)^2} \tag{3.9}$$

are the standard score, sample mean, and sample standard deviation, respectively.

Spearman correlation - The Spearman correlation coefficient is the nonparametric version of the Pearson correlation coefficient. Spearman's correlation coefficient, measures the strength of association between two ranked variables. Spearman's rank correlation coefficient, ρ, is computed from these:

$$\rho = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2 \sum_{i=1}^{n}(y_i - \bar{y})^2}} \tag{3.10}$$

As the Pearson's correlation coefficient, the Spearman's correlation coefficient also can take a value between +1 and -1.

Cosine similarity - is a measure of similarity between two vectors by calculating the cosine of the angle between them. For 0^o the cosine is 1, and less than 1 for any other angle. It measures the similarity of two vectors by calculating its orientation: two vectors with the same orientation have a Cosine similarity of 1, the value decreases to 0 as the vectors differ in orientation, and two opposed vectors have a similarity of -1, independently of their magnitude. Given two vectors X and Y, the cosine similarity, $cos(\theta)$, is derived from the dot product and magnitude as follows:

$$similarity = cos(\theta) = \frac{X \cdot Y}{\|X\|\|Y\|} = \frac{\sum_{i=1}^{n} X_i Y_i}{\sqrt{\sum_{i=1}^{n}(X_i)^2}\sqrt{\sum_{i=1}^{n}(Y_i)^2}} \tag{3.11}$$

3.6.2 Cross-Correlation

To compute the similarity between two vectors an obvious algorithm is the cross-correlation. Cross correlation is mostly used in signal processing and is a measure of similarity of two waveforms as a function of a time-lag applied to one of them. This method has applications in pattern recognition, digital signal processing and others [35].

Cross-correlation is mostly used to compute similarity between signals in the time domain. l, called lag, is the time-shift between the two signals. A sequence $y(n)$ is said to be shifted by l samples with respect to the reference $x(n)$. Cross-correlation is defined as follows:

$$r_{xy}(l) = (x \star y)(l) = \sum_{n=-\infty}^{\infty} x(n)y(n-l) = \sum_{m=-\infty}^{\infty} x(m+l)y(m) = r_{yx}(-l) \tag{3.12}$$

for convenience it can be normalized as follows:

$$\rho_{xy}(l) = \frac{r_{xy}(l)}{\sqrt{r_{xx}(0)r_{yy}(0)}} \tag{3.13}$$

Considering the sequences x and y to be similar and differing only by a shift. Cross-correlation can be used to compute how much y must be shifted to be identical to x. This method slides the sequences along each other and calculates the integral of the product at each shift. The value of $(x \star y)$ reaches a maximum when the two sequences are more similar, this happens because the product of the two functions is higher when the peaks are aligned.

For machine vision applications, several authors [36, 37] suggest the use of cross correlation in frequency domain. For this the discrete Fourier transform (DFT) is used. The sequence of N complex numbers $x_0, x_1, \ldots, x_{N-1}$ is transformed into an N-periodic sequence of complex numbers:

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-j2\pi kn/N}, 0 \leq k \leq N-1. \tag{3.14}$$

the Inverse discrete Fourier transform is given by:

$$x(n) = \frac{1}{N} \sum_{n=0}^{N-1} X(k) e^{j2\pi kn/N}, 0 \leq n \leq N-1. \tag{3.15}$$

Being $\mathcal{F}$ the Fourier transform operator, and $\mathcal{F}\{x\}$ and $\mathcal{F}\{y\}$ are the Fourier transforms of x and y, respectively. Then, similarly to convolution theorem:

$$\mathcal{F}\{x \star y\} = \mathcal{F}\{x\} \cdot \mathcal{F}\{y\} \tag{3.16}$$

where $\cdot$ denotes point-wise multiplication. By applying the inverse Fourier transform $\mathcal{F}^{-1}$, we can write:

$$x \star y = \mathcal{F}^{-1}\{\mathcal{F}\{x\} \cdot \mathcal{F}\{y\}\} \tag{3.17}$$

The fast Fourier transform (FFT) is an algorithm to compute the discrete Fourier transform (DFT) and its inverse. An FFT computes those transformations by factorizing the DFT matrix into a product of sparse (mostly zero) factors. Such factorization can result in significant savings in the computational complexity. With the Fourier transforms one can use the property from equation 3.17 to reduce the cross-correlation to simple products, this reduces significantly the computational complexity of the whole process of cross-correlation. Several FFT algorithms are described in detail in [35].

3.6.3 Algorithms

To use the similarity coefficients methods to compute the displacement between two snapshots it is necessary an algorithmic procedure. The algorithm used is as illustrated in Figure 3.14. First a line scan is captured, then the second line; the correlation coefficient is computed; a shift is applied to the samples and the correlation is computed again, this step is repeated k times until the lines reach 60 % overlap, as suggested by Kálmán [18]; at last the maximum correlation is found and its index corresponds to the displacement between the samples. This procedure is repeated for each new line scan.

For example, in Figure 3.15 are shown two successive snapshots, the plots represent the pixel intensity of each image. After applying the algorithm there is a correlation coefficient for each shift, as shown in Figure 3.16, and the displacement of the two snapshots can be found by locating the maximum of the plot. In this example the

displacement was 80 pixels and the correlation coefficient at that peak was 0.9028. The procedure is then repeated for the following lines.

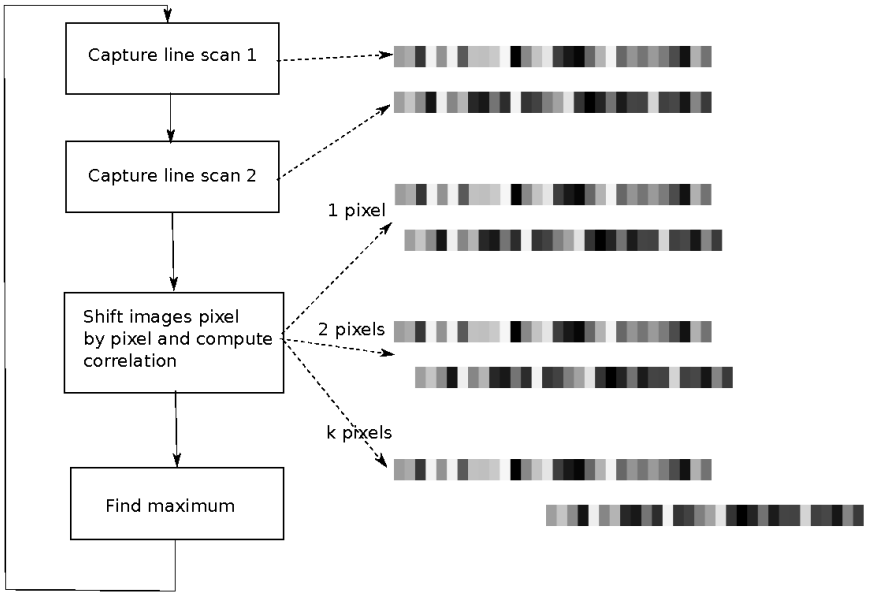


Figure 3.14: Illustration of the algorithm using the similarity coefficients. (Adapted from [18].)

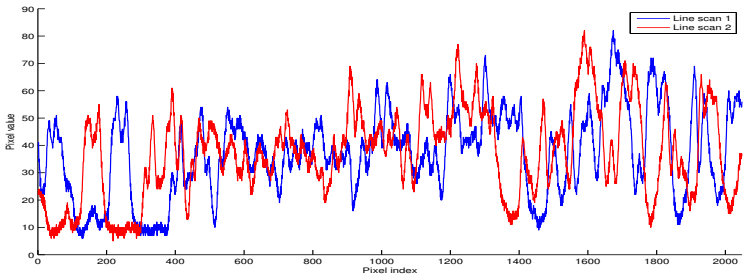


Figure 3.15: Two successive line scans with 80 pixels displacement.

For the cross-correlation the process is simpler, the algorithm consists in performing a *sliding dot product* between the two image samples. This method is already implemented in MatLab, using FFT, by the xcorr() function. On Figure 3.17 there is an example of applying cross-correlation to the line-scans from Figure 3.15. If the lines have N number of pixels, the result will be a $2N-1$ size vector of correlation coefficients, in this case the lines have 2048 pixels and the result from cross-correlation is a 4095 length vector with the correlation coefficients for each shift. In this example the displacement calculated was also 80 pixels with a correlation value of 0.9305.

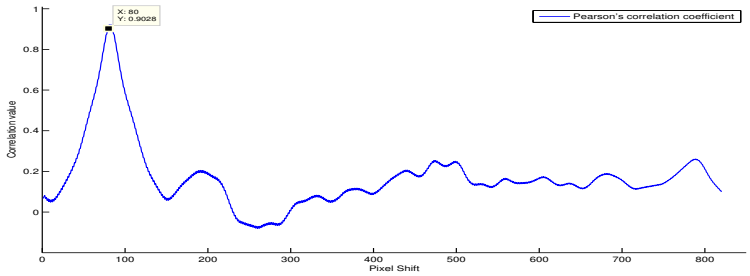


Figure 3.16: Pearson's correlation coefficients between the two line scans.

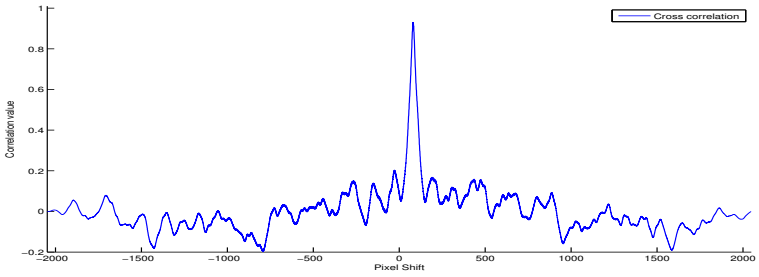


Figure 3.17: Cross-correlation coefficients between the two line scans.

Chapter 4

Experimental Procedure

This chapter describes the experimental procedure to test the algorithms and techniques: first, the materials and setup used, then the details and the assumptions made in the experimental procedure and the variables to be studied.

4.1 Equipments

The camera used on the experimental procedure was an industrial line scan camera. The model was P2-4x-04K40 (7 μm) from Teledyne DALSA (Figure 4.1).

Figure 4.1: Line scan camera [38].

The specifications of the camera are shown in the table 4.1. The hardware interface is Camera Link, a high-speed serial standard; this imposed some limitations and the acquisition had to be done on Windows workspace due to the lack of drivers in Linux. The software used was the SaperaLT, provided by Teledyne DALSA, which was used to capture the frames to a video file to posterior computational processing. This camera model is high speed and is highly configurable; it has the ability to operate from 350 Hz to 36000 Hz line rate, and with a resolution from 128 to 4096 pixels [38].

The optics are the model Rodagon 50 mm f/2.8 and its specifications are shown in table 4.2.

To prove the ability of this method, it is necessary to make experiments with several types of movements. The movements were made using an industrial robot to ensure repeatability. The robot is a six degrees of freedom anthropomorphic robot $FanucM-6iB6s$ capable of manipulating of up to 6 kg objects (Figure 4.2). The robot is well calibrated and has a repeatability of $\pm 0.08mm$ [40], thus its movement was taken as ground truth. The robot gives its position data through a TCP/IP connection. A

Table 4.1: General Camera Specifications [38].

Feature	Specification
Resolution	4096 x 1
Total Data Rate	160 MHz
Max. Line Rate	36 kHz
Pixel Size	7 μm
Output Format	Camera Link Base
Responsivity	up to 38 DN(nJ/cm2) @ 10 dB gain

Table 4.2: General Optics Specifications [39].

Feature	Specification
Lens	50 mm f/2.8
Maximum film format	24x36 mm
Scale range	2-15x
Smallest aperture	16
Flange focal length	43.5 mm
Max. diameter	50.0 mm
Flange to rear edge	13.0 mm

C++ client application was developed to acquire the data from the robot server. This application also calculates the time-stamps; later, the velocity was calculated with this information. The acquisition rate was limited to 5 Hz due to the communication settings in the robot server. The client application asks the robot server for the current position but the answer from the server has a delay, which is not constant. Because of this, the acquisition rate was reduced to 5 Hz; at 5 Hz there are no delays in the robot response, and thus the measurements are consistent.

Figure 4.2: FANUC M-6iB/6S [40].

To measure the orientation of the movement two IMUs were used. The IMUs are 9 degrees of freedom, composed by a gyroscope, an accelerometer and a magnetometer, each of which has 3 degrees of freedom. In the context of this project only the gyroscope data

was used. The model is POLOLU - MinIMU9DOF (Figure 4.3), and the specifications of the sensors are as follows:

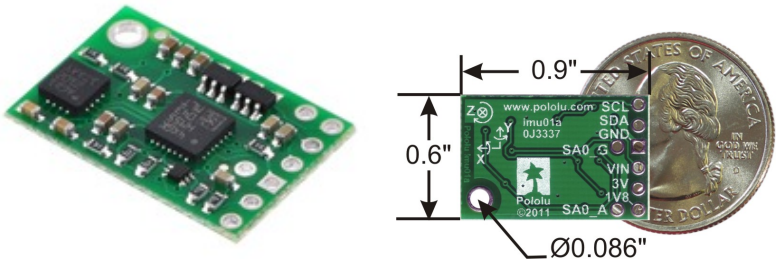


Figure 4.3: POLOLU - MinIMU9DOF, 9 DOF inertial measurement unit [15].

- POLOLU - MinIMU9DOF v2 [15]
    - ∗ Gyroscope 3 degrees of freedom - L3GD20;
        - · Works up to ±2000 °/s;
        - · Resolution up to 16bit;
        - · Sensitivity up to 8.75 mdps/LSB;
        - · Frequency up to 760 Hz;
    - I2C communication;
    - Possibility of powering at 3.3V-Vdd or 5V-Vin.

These IMUs are part of the work developed at the laboratory by Rafeiro [28]. In is work, Rafeiro, developed an IMU network composed by a microcontroller which makes the acquisition from the IMUs and sends it to a computer. In the figure 4.4) there is the scheme of the connections between the IMUs and the computer. The microcontroller is an Arduino Uno and it deals with the I2C communication with the IMUs.

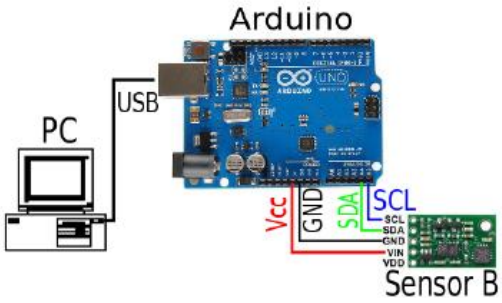
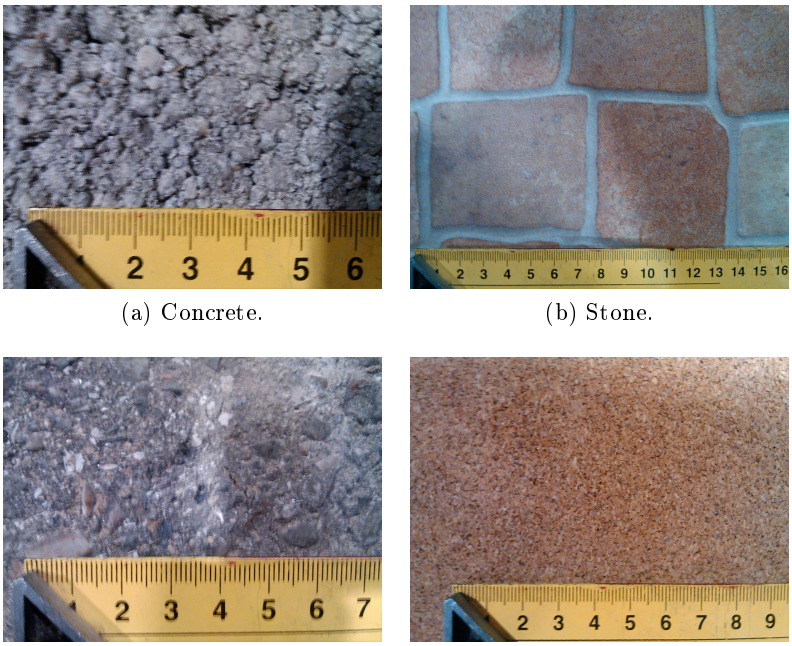


Figure 4.4: Scheme of the connections PC - IMU [28].

4.2 Materials and Samples

To test the method in different textures, four samples of "ground" were used. With these samples it was possible to test the usability of the method in different texture



(a) Concrete. (b) Stone.

(c) Asphalt. (d) Cork.

Figure 4.5: Ground samples used in the experiments.

sizes, regularity and color. The samples used were concrete, stone, asphalt and cork, as shown in Figure 4.5.

In all the experiments, the illumination was controlled to ensure repeatability throughout the day. Sunlight has been blocked and artificial light from two 500 W spotlights were used.

4.2.1 Experimental Setup

The first idea was to fix the camera on the robot, and then, reproduce movements with the camera faced to the ground. This was not possible as the camera support was not strong enough to deal with the accelerations and decelerations which caused vibration. To solve this problem, the camera was mounted stationary on a tripod and the ground materials were fixed on the robot to reproduce the movements. The setup used in the experimental procedure is illustrated in Figure 4.6. A plastic board was fixed on the robot arm, on which were fixed the sample materials, the calibration pattern and the gyroscopes. The camera was installed on a tripod with yaw, pitch, and roll adjustments. The illumination was directed to the material in a way to avoid shadows.

With this setup it was possible to ensure repeatability throughout the experiments. The procedure for each experiment was as follows:

- Experimental Procedure
  - Camera positioning - The camera is positioned at the working distance from the robot, faced to the calibration pattern;

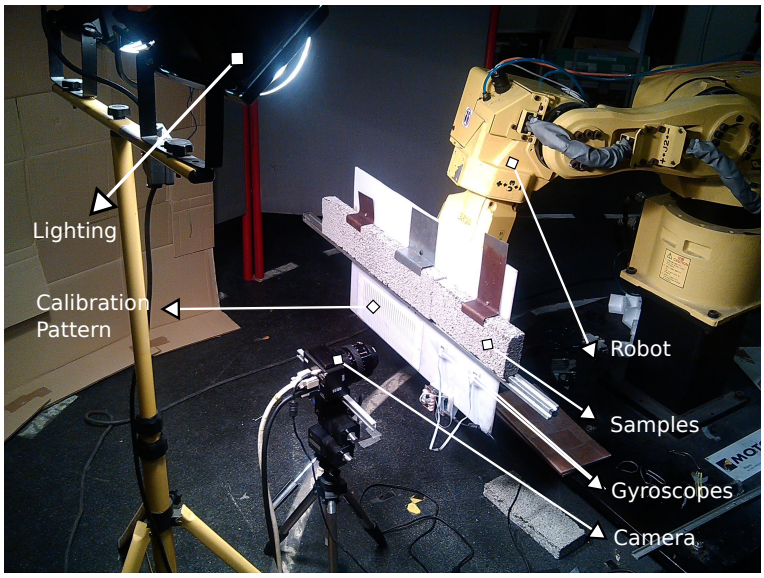


Figure 4.6: The experimental setup on the industrial robot.

- Focus adjustment - The optics focus is adjusted until the lines are clearly visible;
- Camera calibration - The angle of the camera is adjusted until the distance between the pairs of lines is equal, after this the camera will be horizontal to the movement and the field of view is determined. After this step the camera will not be moved anymore;
- The ground materials are installed in the robot and the working distance is adjusted to compensate the thickness of the material. The distance is compensated adjusting the robot position;
- The measurements are made with the different velocities and materials;

This procedure was repeated to the other distances. The distance between the camera and the material was measured from the tip of the optics to the face of the material, as illustrated in Figure 4.7. In each measurement a video with all the frames is recorded and the positions from the robot are stored on a text file for posterior computational processing. For the circular movements the data from the gyroscopes is also stored in a text file, synchronized with the robot data.

4.3 Movements

To prove the ability of this method, it is necessary to make experiments with several types of movements, to simplify the problem, those experiments were divided in four phases. All the movements were made with a length of 500 mm.

The movements were repeated for each sample material. The velocities used in the experiments were 100, 500 and 1000 mm/s. It was not possible to go higher than 1000 mm/s due to the fact that the base of the robot was not firmly fixed to the ground. So, for safety concerns, the maximum velocity was limited.

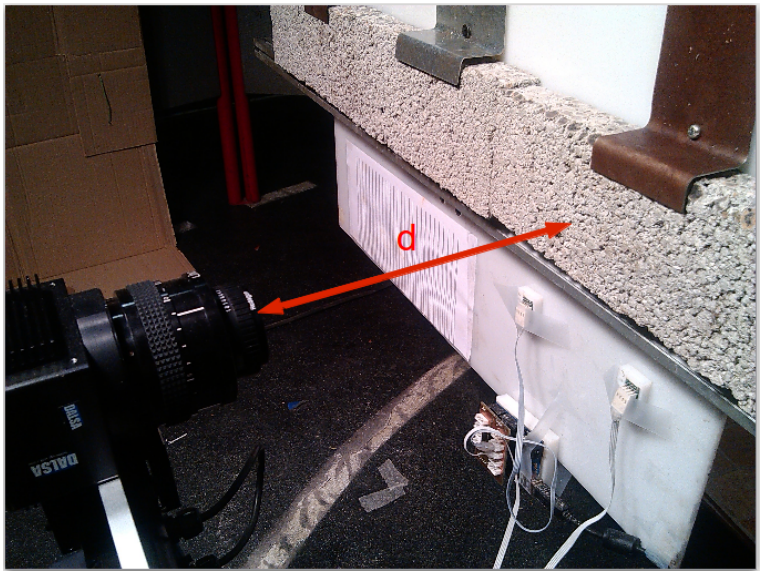


Figure 4.7: Illustration of the working distance.

4.3.1 Parallel movement

The first movement of the experiments was horizontal straight movements with constant distance to the ground and constant velocity. The movement was always parallel to the axis of the line scan sensor; to ensure this, the camera was calibrated as stated in section 4.4. This movement was made at the distance of 200 mm and 300 mm to the ground. The experiment was repeated for 100, 500 and 1000 mm/s for each distance. On Figure 4.8 is a representation of the lines captured, each rectangle is a line scan; in case the angle θ is zero degrees, the lines captured will overlap in a straight line.

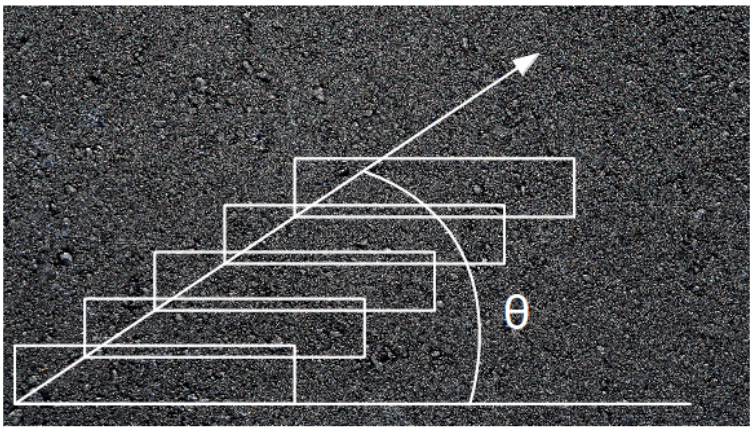


Figure 4.8: Illustration of the concept of the diagonal movement.

4.3.2 Diagonal movement

The diagonal movements were similar to the parallel on, but with an angle. This is important to study the effect of having a perpendicular component on the movement.

This can happen when the vehicle is sliding or when the sensor is poorly installed. The distance and velocity was also constant. This movement was made at the distance of 200 mm and 300 mm to the ground. The experiment was repeated for 0, 2.5, 5 and 10 degrees of angle, and each angle repeated for 100, 500 and 1000 mm/s. This movement is illustrated on Figure 4.8, showing that sections of the lines will overlap due to its width.

4.3.3 Circular movement

The third movement was circular and aims to study the ability of this system to deal with rotations on the successive frames. The movements were made to simulate a curve of a vehicle with Ackermann steering. This movement was made at the distance of 200 mm and 300 mm to the ground. The experiment was repeated for 2, 5, and 10 meters of circle diameter and each of which repeated for 100, 500 and 1000 mm/s.

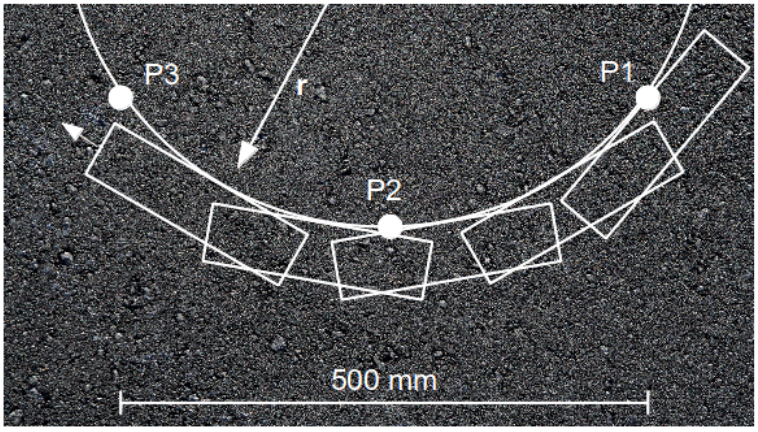


Figure 4.9: Illustration of the concept of the circular movement.

4.3.4 Oscillatory movement

The last movement was made with changing distance of the camera to the ground to study the effect of changing the focal length and magnification of the lens, this is a situation that happens in real world environments like having an irregular pavement or having fluctuations on the vehicle suspension. The movement is as illustrated on Figure 4.10; it is a sinusoid with amplitude of 25 mm.

4.4 Camera Calibration

Camera calibration is a very important step, it is necessary to get the relationship between the object coordinates and the image coordinates. With known parameters of the camera model, including internal parameters and external parameters, it is possible to determine this transformation. Calibration is used to determine these parameters. Camera matrix is one of the important internal parameters and position and orientation of the sensor to the world coordinate system are some of the important external parameters [41].

In order to calculate the ground displacement after displacement between successive frames is necessary to know the relationship between the image size and the field of view

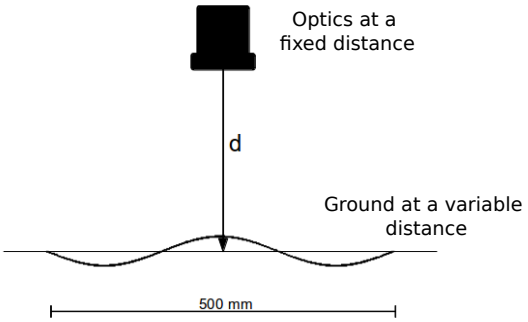


Figure 4.10: Illustration of the concept of the oscillatory movement.

size. By knowing this relation makes is possible to calculate the real ground displacement at a given distance from the ground. The field of view is obviously dependent of the distance of the camera to the ground and for a better result it is essential to know this distance. Therefore, in a real implementation, a method of estimating this distance should be available.

The experimental procedure proposed will require that the sensor is parallel to the movement; the image, being 1-D, brings difficulties to ensure that. To guarantee that the sensor is parallel to the movement, a calibration method suggested by [41, 42] is used. The method implies the use of a pattern of vertical parallel lines crossed with diagonal lines, parallel to each other (figure 4.11).

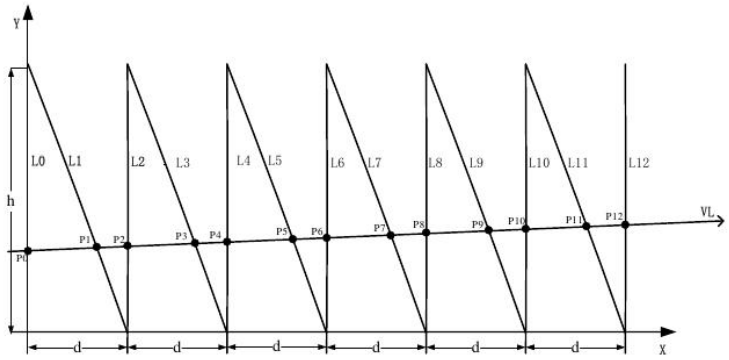


Figure 4.11: Pattern used for line scan cameras calibration [41].

With this configuration, it is easy to see that when the camera is completely horizontal the spaces between the pairs of lines will have an equal distance, and when the camera is oblique these spaces will have a tendency to increase or decrease depending on the orientation of the sensor.

In Figure 4.12 it is possible to see a image of the process of calibration, the image shown here is the representation of 480 repeated lines since the camera has not moved. The camera is adjusted until all the distances between the pairs of lines are equal. Once that is accomplished, the camera sensor is horizontal to the pattern and therefore horizontal to the movement. To automate the process of calibration, a Matlab script was created which calculates the distance between the pairs of lines with precision. In the

Figure 4.12 the distances are in pixels.

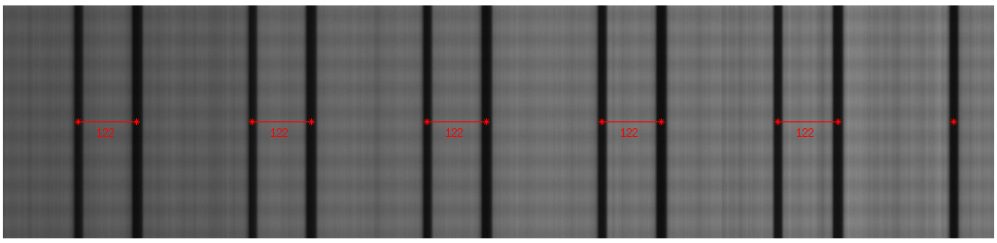


Figure 4.12: Image taken during the process of calibration.

4.5 Gyroscope's Calibration

Typically, gyroscopes show a systematic error in their measurements which can induce errors that can be harmful to movement estimation. These systematic errors are usually named bias. This process of calibration has the objective of determining the difference between the measurements and the real value. This process can be used further to compensate the bias and ensure a better precision.

To calibrate the gyroscopes, a static position was imposed to them and then the measurements were done. The mean difference compared to the reference, which is 0 °/s in this case, was calculated to be used as compensation. To improve the precision even more, two gyroscopes were used and the final value to consider was the mean of the two already calibrated gyroscopes.

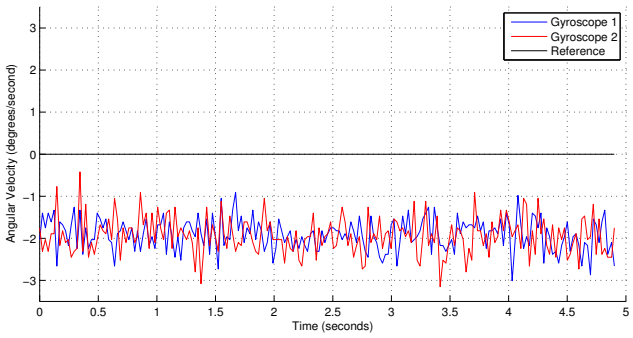


Figure 4.13: Raw data from two gyroscopes on stationary position; Z axis at 40 Hz.

As illustrated in Figure 4.13, the gyroscopes showed biases from the reference value (0 °/s). Figure 4.14 shows the result of the offset correction. Figure 4.15 show the final result of the offset correction and average of the two gyroscopes. Finally, in Table 4.3 the results for standard deviation and mean values of the gyroscope's measurements are shown. It is clear that using two gyroscopes with offset correction decreases the deviation of the readings; also the mean value has a residual difference.

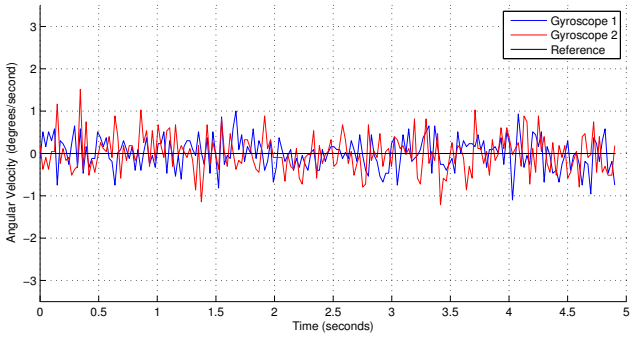


Figure 4.14: Data from two gyroscopes with offset compensation; Z axis at 40 Hz.

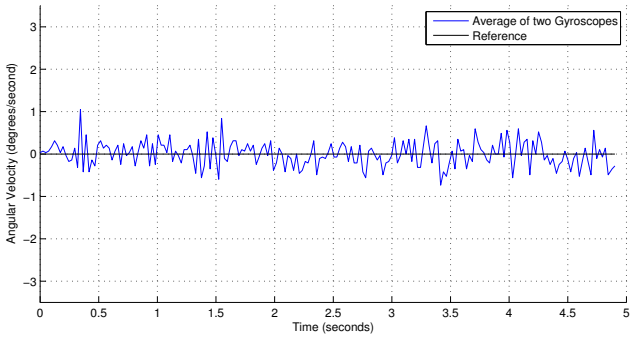


Figure 4.15: Result of the average of two calibrated gyroscopes; Z axis at 40 Hz

Table 4.3: Standard deviation and mean value from the gyroscopes - Z axis in stationary position.

		Standard Deviation ($^o/s$)	Average Value ($^o/s$)
Raw data	Gyroscope 1	0.3760	-1.9124
	Gyroscope 2	0.4391	-1.9362
With offset correction	Gyroscope 1	0.3760	-0.0877
	Gyroscope 2	0.4391	-0.0533
Average of the two gyroscopes		0.2932	-0.0805

Chapter 5

Results and Discussion

This chapter describes and discusses the experimental results and its discussion. First, the results of all the movements are presented. Next, a simulation of higher velocities is made, using the method of sub-sampling. And at last the results are discussed.

The optics magnification was kept fixed through the experiments. At the distance of 200 mm the field of view was 50.67 mm and at the distance of 300 mm the field of view was of 77.65 mm.

The movements were captured by the camera and the proprietary software saves it in .avi video format. The videos were then processed in matlab, the frames were extracted and merged into an only image with all the frames of the movement. An example of these images is shown in Figure 5.1. Figure (a) shows a movement using cork at distance 200 mm and velocity 500 mm/s; figure (b) shows a movement using concrete at distance 200 mm and velocity 500 mm/s.

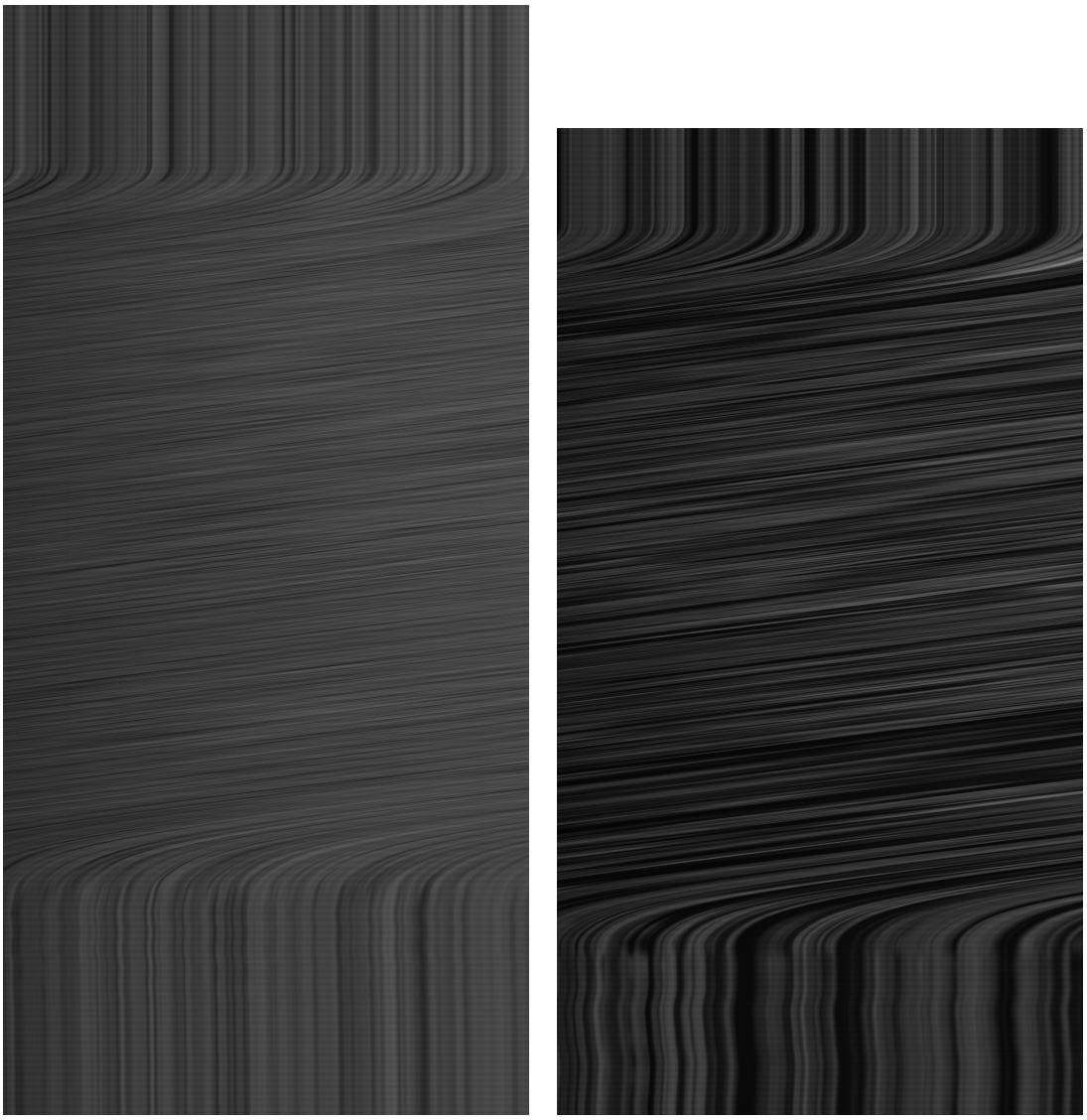
Wile performing the first experiments it was found that the software was not working properly and it was not possible to capture images in full resolution (1×4096 pixels), so the image acquisition was made in half resolution (1×2048 pixels). The reason for this event is unknown and it might have been a malfunctioning driver.

Overall, the best results occurred using cross-correlation. In comparison with the methods of similarity, using cross-correlation were obtained higher values of coefficient of correlation. Also, the cross-correlation algorithm using FFT was, in average, ten times faster than the algorithm with the similarity coefficient methods. For those reasons, the following results are the ones made with cross-correlation.

5.1 Experimental Results

In this section there will be presented the results of the experiments. The images taken with the camera were processed with the algorithm shown on section 3.6 using cross-correlation.

This section show the results using the algorithm described in section 3.6.3. After these results it was introduced an improvement to the algorithm, that will be explained in section 5.2; the analysis was then repeated using that algorithm. Table 5.1 shows the plan of the results with the numbering for each subsection.



(a) Cork. (b) Concrete.

Figure 5.1: Illustration of the final image of the whole movement. Each line scan captured is merged vertically into one only image with all the scans made along the movement. These images are of size $n \times 2048$, where n is the number of scans.

Table 5.1: Page numbering of the results with the simple algorithm and with the dynamic sub-sampling.

	Simple analysis	Dynamic sub-sampling
Parallel	39	54
Diagonal	42	57
Circular	43	58
Oscillatory	49	64
Discussion	51	66

5.1.1 Parallel Movement

For the parallel movement the velocity estimated was highly similar for all of the materials. The following figures, show the plots for the velocity calculated with the data collected from camera. Each figure shows, also, the velocity plot from the robot (ground truth). The measurements presented were made on concrete and it should be emphasized that the plots for the other materials are similar.

Figure 5.2 shows a big offset between the velocity calculated by the camera sensor and the ground truth. This is due to the fact that the correlation algorithm only detects a pixel displacement that is a multiple of 2. The pixel displacement calculated here was 2 and it is equivalent to 123.71 mm/s.

Figure 5.3, 5.4 and 5.7 show consistent measurements from the image sensor.

Figure 5.5 show high fluctuations of the velocity, between 0 and 189.5735 mm/s. Here the successive frames were so similar to each other that the computed displacement oscillated. At some cases the displacement calculated was 2 pixels and at others had the value of 4, that caused the velocity fluctuation.

Figure 5.6 shows an offset on the velocity measured by the sensor and also some fluctuations. Again, the offset was caused because the pixel displacement is always an integer and converting it to velocity induces error.

Table 5.2 show the velocity errors for all the movements with all the materials. For the velocity it was calculated the average error and the maximum error using the robot as reference. It was also calculated the error for the total distance traveled. The results show that the tendency of the error is the same to all the materials.

As expected, the error for 100 mm/s at the distance of 200 mm is high due to the offset calculated. The same happened for 500 mm/s at the distance of 300 mm. The higher error values were, as expected, for 100 mm/s at 300 mm since the velocity measurements had high fluctuation. At the distance of 300 and velocity of 1000 mm/s there was a small offset and that caused an error around 5%. For the other cases the results were satisfactory with an error around 1%.

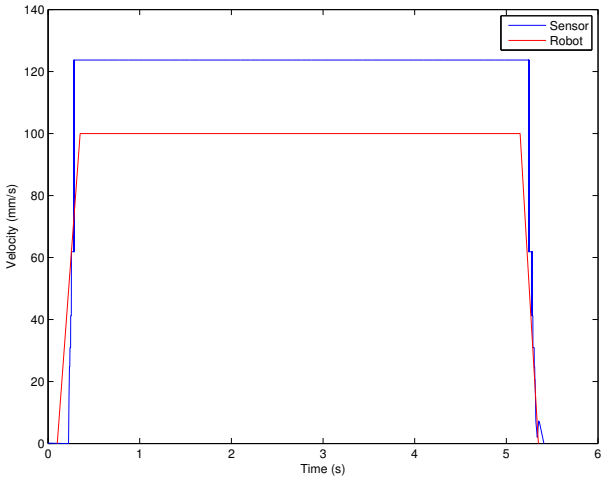


Figure 5.2: Parallel movement - Velocity plot - 100 mm/s at distance of 200 mm.

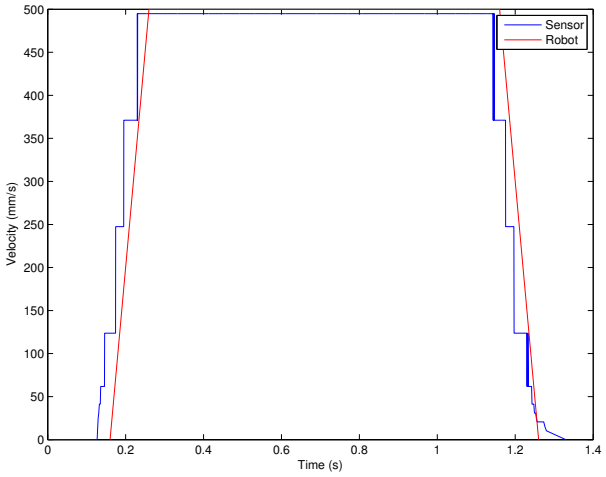


Figure 5.3: Parallel movement - Velocity plot - 500 mm/s at distance of 200 mm.

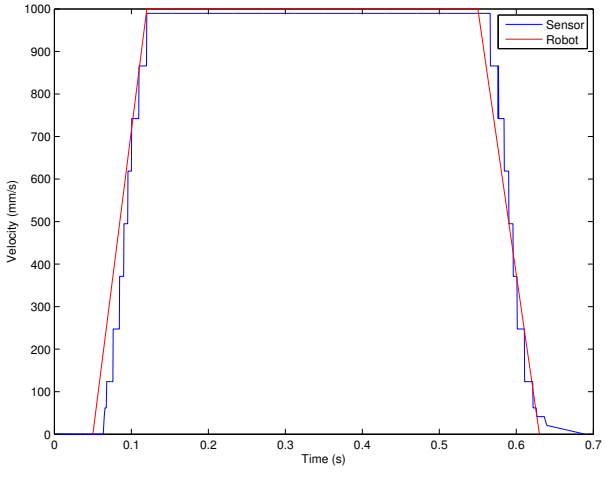


Figure 5.4: Parallel movement - Velocity plot - 1000 mm/s at distance of 200 mm.

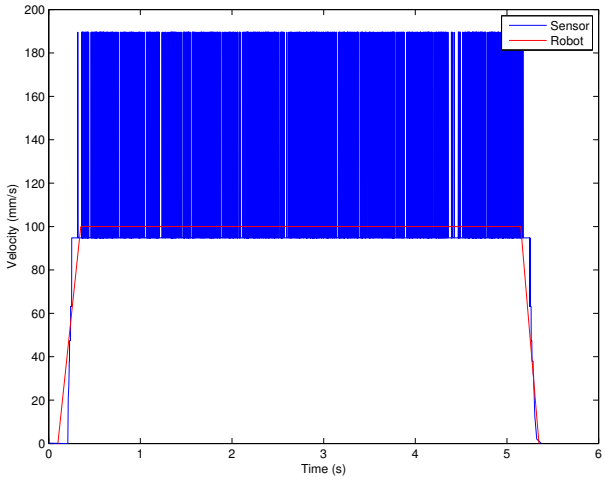


Figure 5.5: Parallel movement - Velocity plot - 100 mm/s at distance of 300 mm.

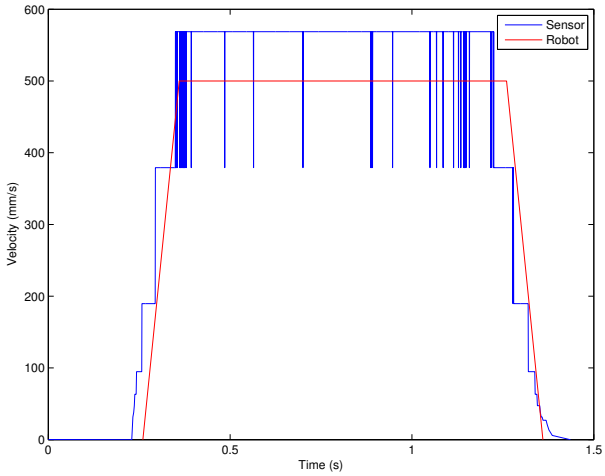


Figure 5.6: Parallel movement - Velocity plot - 500 mm/s at distance of 300 mm.

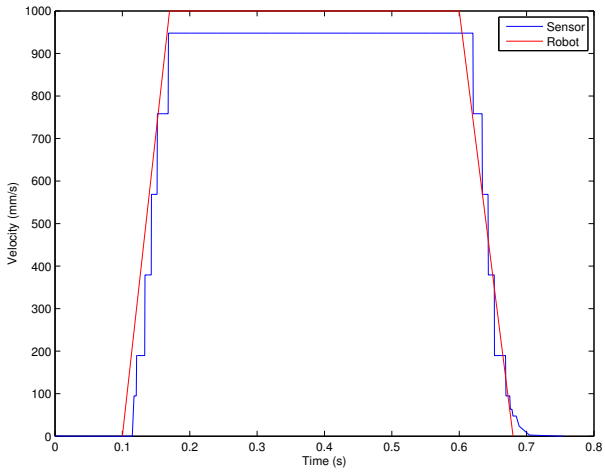


Figure 5.7: Parallel movement - Velocity plot - 1000 mm/s at distance of 300 mm.

Table 5.2: Parallel movement - Summary of error results.

		Concrete		Cork		Stone		Asphalt	
Distance Separation (mm)	Target Velocity (mm/s)	Avg. Vel. (%)	Dist. Travel. (%)	Avg. Vel. (%)	Dist. Travel. (%)	Avg. Vel. (%)	Dist. Travel. (%)	Avg. Vel. (%)	Dist. Travel. (%)
200	100	23.71	24.20	23.71	24.00	23.40	22.92	23.71	24.00
	500	1.03	0.50	1.03	0.47	1.03	0.65	1.03	0.50
	1000	1.03	0.76	1.03	0.75	1.03	0.77	1.03	0.75
300	100	62.74	49.26	13.53	5.83	22.37	12.21	53.12	49.20
	500	13.57	11.65	13.16	11.17	12.08	10.09	13.20	11.32
	1000	5.21	4.68	5.21	4.76	5.21	4.67	5.21	4.51

5.1.2 Diagonal Movement

For the diagonal movement the velocity plots were highly similar to the parallel one. In nature the two movements are very similar, only the diagonal has an angle applied to it.

Table 5.3 shows the average velocity and total distance relative error for all the movements with all the materials at the distance of 200 *mm*. Table 5.4 shows the results for a distance of 300 *mm*. It was calculated the average error and the maximum error using the robot as reference.

Here, it is possible to see that the tendency of the error is the same as for the parallel movement. In fact, the results were similar in the two cases, being even equal in most of the movements. This is an evidence that this measuring method is robust even when the image sensor has a rotation relative to the line of movement.

Table 5.3: Diagonal movement - errors for a separation of 200 *mm*.

		Concrete		Cork		Stone		Asphalt	
Angle (°)	Target Velocity (mm/s)	Avg. Vel. (%)	Dist. Travel. (%)	Avg. Vel. (%)	Dist. Travel. (%)	Avg. Vel. (%)	Dist. Travel. (%)	Avg. Vel. (%)	Dist. Travel. (%)
2.5	100	23.71	24.37	23.71	24.27	23.32	23.10	23.71	24.37
	500	1.03	0.21	1.03	0.32	1.03	0.08	1.03	0.23
	1000	1.03	0.07	1.03	0.03	1.03	0.10	1.03	0.10
5	100	23.71	25.35	23.71	24.66	22.97	22.42	23.71	25.20
	500	1.03	0.30	1.03	0.27	1.03	0.01	1.03	0.31
	1000	1.03	0.53	1.03	0.02	1.03	0.15	1.03	0.43
10	100	23.71	19.84	23.71	26.13	23.32	24.18	23.71	18.20
	500	1.03	0.86	1.03	0.98	1.03	0.90	1.03	0.76
	100	1.06	0.34	1.02	0.67	1.03	0.61	1.03	0.40

Table 5.4: Diagonal movement - errors for a separation distance of 300 mm.

		Concrete		Cork		Stone		Asphalt	
Angle (°)	Target Velocity (mm/s)	Avg. Vel. (%)	Dist. Travel. (%)	Avg. Vel. (%)	Dist. Travel. (%)	Avg. Vel. (%)	Dist. Travel. (%)	Avg. Vel. (%)	Dist. Travel. (%)
2.5	100	67.04	52.56	16.19	7.48	23.45	28.36	56.12	40.21
	500	13.48	12.38	11.59	10.33	9.51	8.56	13.20	12.46
	1000	5.21	3.94	5.21	3.97	5.21	4.02	5.21	3.90
5	100	51.21	38.33	13.60	6.36	0.81	1.74	49.30	47.65
	500	3.49	4.30	10.27	9.32	10.93	9.95	3.20	4.12
	1000	5.21	3.36	5.21	4.00	5.21	3.98	5.21	3.42
10	100	43.78	30.20	9.10	4.90	0.53	0.98	39.25	37.24
	500	2.79	1.32	6.05	4.12	3.69	4.39	2.54	1.23
	100	5.21	3.47	5.21	3.27	5.21	3.32	5.21	3.12

5.1.3 Circular Movement

Figures 5.8, 5.9, 5.10, 5.11, 5.12, 5.13 show the generic velocity plots for the circular movement at the various velocities and distances. It shows also the trajectory made by the robot versus the one estimated by the camera and gyroscopes.

Tables 5.5, 5.6, 5.7, 5.8, 5.9, 5.10, 5.11, 5.12 show the error and final position deviation for all the experiments. The tables show the average velocity error and total distance error, they also show the X and Y coordinates deviation and the euclidean distance between the end path points estimated and the ground truth. The final position of the robot was $(x, y) = (500, 0)$ and the X, Y deviations are calculated as the difference between these and the ground truth.

In general, the estimated trajectories made with the camera and gyroscopes showed small deviation from the reference. Except for the experiments at 100 mm/s wish had high errors in the longitudinal velocity estimation.

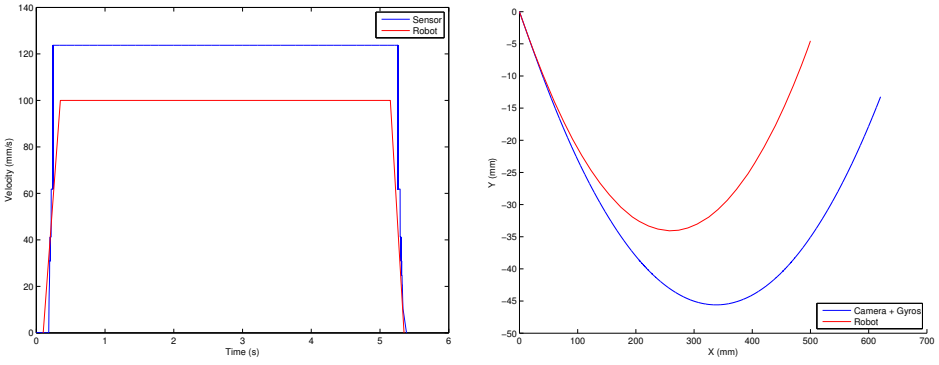


Figure 5.8: Velocity and position plot for velocity of 100 mm/s - diameter of 2 m - at distance 200 mm.

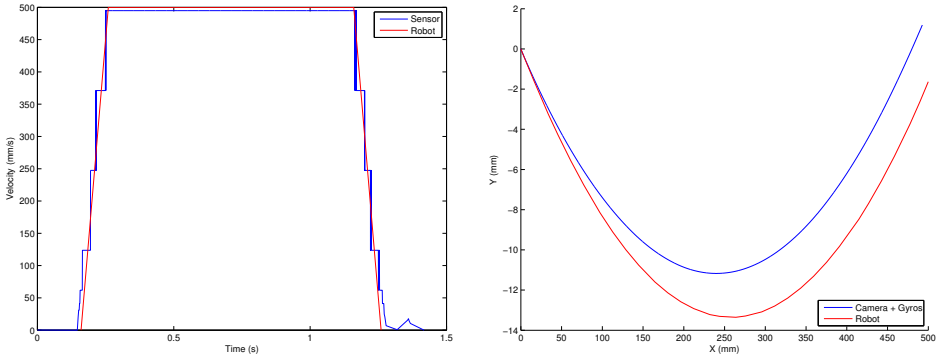


Figure 5.9: Velocity and position plot for velocity of 500 mm/s - diameter of 5 m - at distance 200 mm.

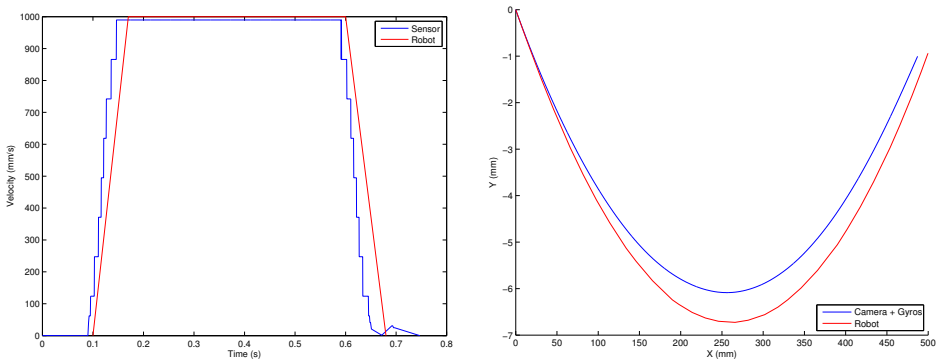


Figure 5.10: Velocity and position plot for velocity of 1000 mm/s - diameter of 10 m - at distance 200 mm.

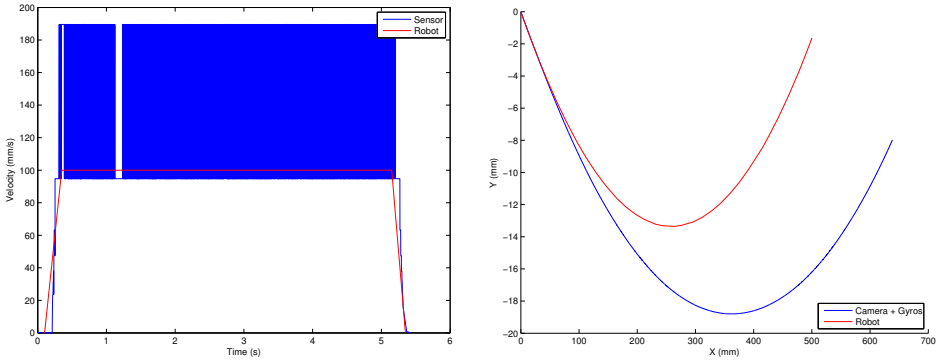


Figure 5.11: Velocity and position plot for velocity of 100 mm/s - diameter of 5 m - at distance 300 mm.

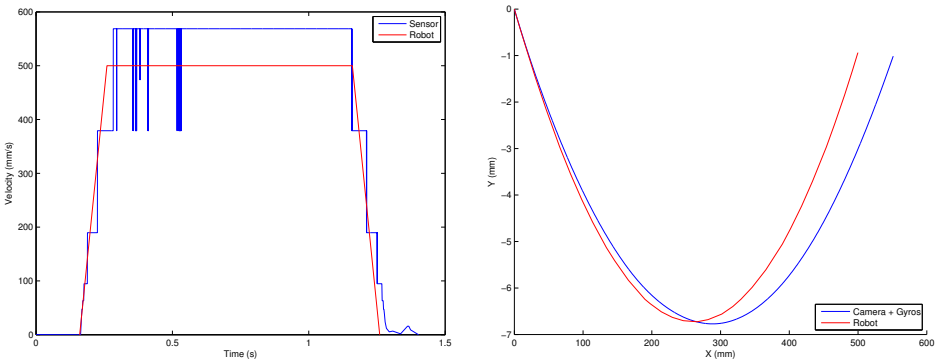


Figure 5.12: Velocity and position plot for velocity of 500 mm/s - diameter of 10 m - at distance 300 mm.

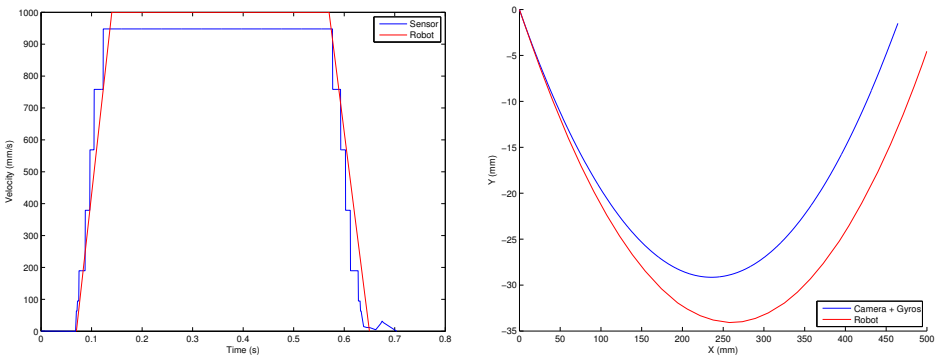


Figure 5.13: Velocity and position plot for velocity of 1000 mm/s - diameter of 2 m - at distance 300 mm.

Table 5.5: Circular movement - errors for concrete at a distance separation of 200 mm.

			Concrete				
			Error		Deviation		
Distance Separation (mm)	Diameter (mm)	Target Velocity (mm/s)	Avg. Vel. (%)	Dist. Travel. (%)	X (mm)	Y (mm)	Euclidean Distance (mm)
200	2	100	23.71	25.34	120.38	13.24	121.10
		500	1.03	0.07	12.34	0.55	12.35
		1000	11.15	8.97	60.69	1.90	60.72
	5	100	23.71	26.65	132.97	0.72	132.97
		500	1.03	0.13	7.36	1.19	7.45
		1000	1.03	0.49	12.84	0.74	12.86
	10	100	23.71	24.12	120.40	16.39	121.51
		500	1.03	0.69	8.21	19.11	20.80
		1000	1.03	1.03	12.83	19.00	22.93

Table 5.6: Circular movement - errors for concrete at a distance separation of 300 mm.

			Concrete				
			Error		Deviation		
Distance Separation (mm)	Diameter (mm)	Target Velocity (mm/s)	Avg. Vel. (%)	Dist. Travel. (%)	X (mm)	Y (mm)	Euclidean Distance (mm)
300	2	100	4.74	3.09	22.92	11.12	25.48
		500	24.17	20.55	115.17	1.10	115.17
		1000	5.21	3.89	35.58	1.50	35.61
	5	100	40.58	27.55	138.34	7.98	138.57
		500	6.02	5.42	19.34	0.42	19.35
		1000	5.21	4.11	31.21	0.63	31.22
	10	100	64.52	51.30	257.22	13.86	257.59
		500	13.24	11.38	51.15	18.99	54.56
		1000	5.21	4.67	31.05	18.97	36.39

Table 5.7: Circular movement - errors for cork at a distance separation of 200 mm.

			Cork				
			Error		Deviation		
Distance Separation (mm)	Diameter (mm)	Target Velocity (mm/s)	Avg. Vel. (%)	Dist. Travel. (%)	X (mm)	Y (mm)	Euclidean Distance (mm)
200	2	100	23.71	25.18	119.52	13.20	120.25
		500	1.03	0.02	12.88	0.02	12.88
		1000	9.82	7.82	55.89	2.13	55.93
	5	100	23.71	24.32	120.82	3.90	120.89
		500	1.03	0.04	7.75	1.25	7.85
		1000	1.03	0.37	13.01	0.81	13.04
	10	100	23.71	24.01	119.93	16.38	121.04
		500	1.03	0.69	9.07	19.19	21.23
		1000	1.03	1.02	13.58	19.04	23.39

Table 5.8: Circular movement - errors for cork at a distance separation of 300 mm.

			Cork				
			Error		Deviation		
Distance Separation (mm)	Diameter (mm)	Target Velocity (mm/s)	Avg. Vel. (%)	Dist. Travel. (%)	X (mm)	Y (mm)	Euclidean Distance (mm)
300	2	100	5.21	3.04	19.92	4.00	20.32
		500	24.17	20.53	115.68	1.01	115.68
		1000	5.21	3.98	36.13	1.58	36.17
	5	100	3.22	0.41	0.63	3.79	3.84
		500	8.39	7.31	44.78	0.06	44.78
		1000	5.21	4.17	31.75	0.60	31.76
	10	100	11.07	4.20	21.11	16.61	26.87
		500	10.43	8.27	34.93	19.35	39.93
		1000	5.21	4.74	31.98	18.99	37.19

Table 5.9: Circular movement - errors for stone at a distance separation of 200 mm.

			Stone				
			Error		Deviation		
Distance Separation (mm)	Diameter (mm)	Target Velocity (mm/s)	Avg. Vel. (%)	Dist. Travel. (%)	X (mm)	Y (mm)	Euclidean Distance (mm)
200	2	100	18.58	15.88	69.39	19.68	72.12
		500	1.04	0.09	11.27	0.14	11.27
		1000	4.38	3.14	32.13	1.80	32.18
	5	100	22.60	21.77	104.14	4.94	104.26
		500	1.03	0.19	7.31	1.27	7.42
		1000	1.03	0.34	12.13	0.76	12.15
	10	100	23.04	22.26	108.09	16.04	109.27
		500	1.03	0.81	8.52	19.23	21.03
		1000	1.03	1.04	13.32	19.06	23.26

Table 5.10: Circular movement - errors for stone at a distance separation of 300 mm.

			Stone				
			Error		Deviation		
Distance Separation (mm)	Diameter (mm)	Target Velocity (mm/s)	Avg. Vel. (%)	Dist. Travel. (%)	X (mm)	Y (mm)	Euclidean Distance (mm)
300	2	100	5.20	3.53	22.59	10.35	24.85
		500	24.17	20.70	115.50	1.02	115.50
		1000	5.21	4.06	36.28	1.41	36.31
	5	100	4.52	3.82	19.76	3.18	20.01
		500	8.40	7.44	44.79	0.22	44.79
		1000	5.21	4.17	31.30	0.55	31.30
	10	100	3.88	3.80	19.99	17.07	26.28
		500	5.90	4.25	15.72	19.22	24.83
		1000	5.21	4.86	32.27	19.01	37.45

Table 5.11: Circular movement - errors for asphalt at a distance separation of 200 mm.

			Asphalt				
			Error		Deviation		
Distance Separation (mm)	Diameter (mm)	Target Velocity (mm/s)	Avg. Vel. (%)	Dist. Travel. (%)	X (mm)	Y (mm)	Euclidean Distance (mm)
200	2	100	23.71	25.34	120.38	13.24	121.10
		500	1.03	0.07	12.34	0.55	12.35
		1000	11.15	8.97	60.69	1.90	60.72
	5	100	23.71	26.65	132.97	0.72	132.97
		500	1.03	0.13	7.36	1.19	7.45
		1000	1.03	0.49	12.84	0.74	12.86
	10	100	23.71	24.12	120.40	16.39	121.51
		500	1.03	0.69	8.21	19.11	20.80
		1000	1.03	1.03	12.83	19.00	22.93

Table 5.12: Circular movement - errors for asphalt at a distance separation of 300 mm.

			Asphalt				
			Error		Deviation		
Distance Separation (mm)	Diameter (mm)	Target Velocity (mm/s)	Avg. Vel. (%)	Dist. Travel. (%)	X (mm)	Y (mm)	Euclidean Distance (mm)
300	2	100	4.74	3.09	22.92	11.12	25.48
		500	24.17	20.55	115.17	1.10	115.17
		1000	5.21	3.89	35.58	1.50	35.61
	5	100	40.58	27.55	138.34	7.98	138.57
		500	6.02	5.42	19.34	0.42	19.35
		1000	5.21	4.11	31.21	0.63	31.22
	10	100	64.52	51.30	257.22	13.86	257.59
		500	13.24	11.38	51.15	18.99	54.56
		1000	5.21	4.67	31.05	18.97	36.39

5.1.4 Oscillatory Movement

For the oscillatory movement the velocity measurements were made at the initial distance of 200 mm, increasing to 225 mm, back to 200 mm and then decreasing to 175 mm. At the middle the robot did a small pause.

The following figures show the velocity plots for each material. Figure 5.14 shows the velocity plot for concrete. An error measuring the velocity can be seen, caused by the changes in magnification. Figure 5.15 shows the result for cork and it is similar to the previous one. Figure 5.16 shows the result for stone, here the error was higher due to the high fluctuation of the values calculated. Stone is a material almost texture-less and the changes in magnification cause blur. This blur induces errors in the velocity estimation. The other materials are less sensitive to the blur as they have bigger grains

on its texture.

Table 5.13 shows the errors for these experiments. The average and maximum error were calculated for velocity and the error for total distance traveled was also calculated.

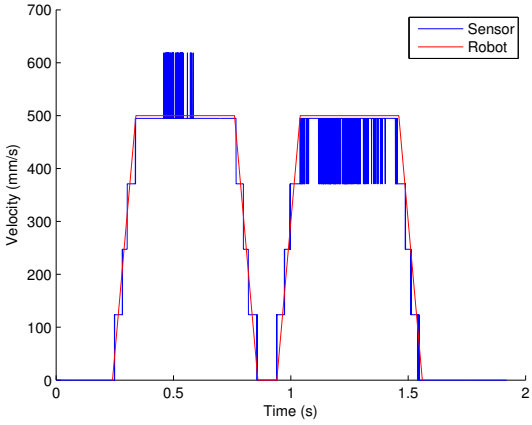


Figure 5.14: Oscillatory movement - Velocity plot - 500 mm/s at distance of 200 mm - concrete

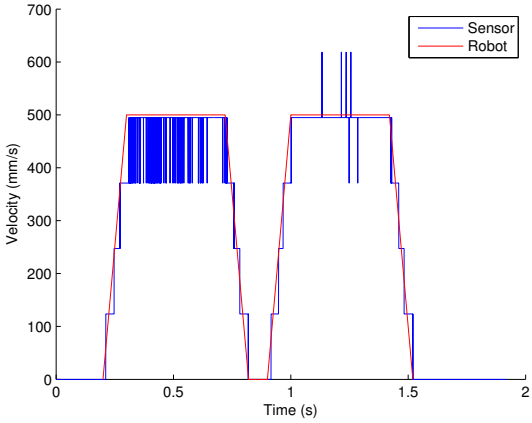


Figure 5.15: Oscillatory movement - Velocity plot - 500 mm/s at distance of 200 mm - Cork.

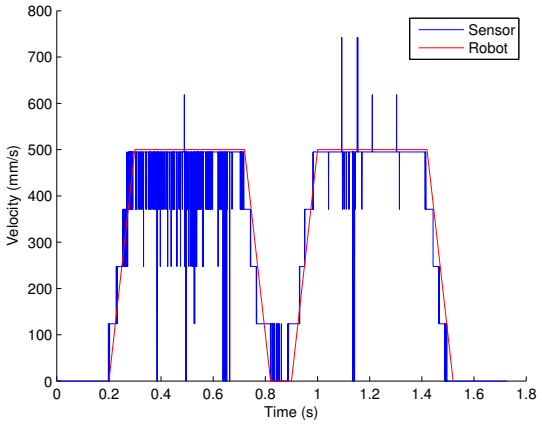


Figure 5.16: Oscillatory movement - Velocity plot - 500 mm/s at distance of 200 mm - stone

Table 5.13: Error Results for oscillatory movement.

	Velocity (mm/s)	Avg. Velocity Error (%)	Total Distance Error (%)
Concrete	500	2.42	0.32
Cork		2.16	0.61
Stone		12.84	8.78
Asphalt		2.53	0.33
Concrete	1000	2.43	0.33
Cork		2.16	0.61
Stone		12.22	8.85
Asphalt		2.61	0.40

5.1.5 Results Discussion

While carrying ou the experiments the sensor showed an overall good result, with measurements similar to the ones made on the robot. Although, there are several conditions in which the precision is affected resulting in increased error.

In all the experiments made at 100 mm/s the error was very high. This is due to the fact that the implemented algorithm only computes pixel displacements that are integers. In this case the real displacement was 0.04 mm. At the distance of 200 mm, the displacement calculated was 2 pixels and that is equivalent to 0.04948 mm.

In all the experiments the pixel displacement calculated was always 2 or a multiple of 2, this can be seen in the example in Figure 5.17 were there is a plot of the pixel displacement correspondent to a movement at 1000 mm/s. One possible explanation for this event might be explained with the taping method of the camera. The camera used in the experiments features a quad tap method to increase the acquisition speed. This method, as described in section 3.4, divides the linear sensor in front/even, front/odd, rear/even and rear/odd. The camera software recorded the images in half resolution and it is possible that the software divided the full image and stored the odd section as the

first line and the even section as the second line, and so on. However, this could not be proved, because when the problem arose, the camera was no longer available to make a more detailed investigation.

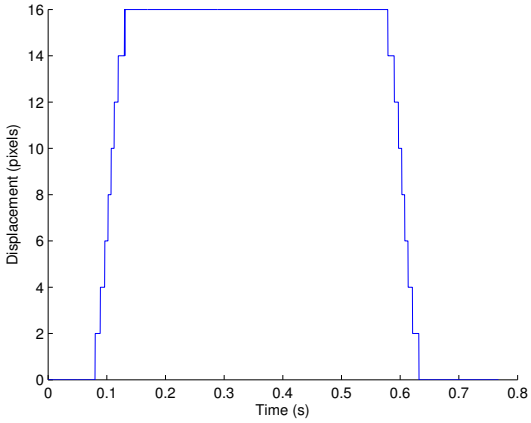


Figure 5.17: Parallel movement - Velocity plot - 100 mm/s at distance of 200 mm with dynamic sub-sampling.

It is also possible to see that successive frames captured can be so similar that the algorithm fails to compute the correct displacement. This fact suggests that the measuring system should have a variable acquisition rate or the algorithm should adapt to the current velocity. Overall, the results were better at the distance of 200 mm. At 300 mm the errors were higher and the only acceptable results were at the velocity of 1000 mm/s. This leads to the conclusion that increasing the field of view reduces the ability to measure small velocities.

The results for the diagonal movements show that having the camera sensor with an angle of up to 10° does not have an effect on the velocity measured. At the distance of 200 mm the errors were the same as the parallel movement. At 300 mm the errors were higher and the only, as previously, and the only matching with the parallel movement was at the velocity of 1000 mm/s. This leads to the same conclusion that increasing the field of view reduces the ability to measure small velocities. These results are important as they prove that the vehicle can have sideways movements and the sensor still being able to measure the velocity accurately. Also, these results show that the camera sensor will operate properly even if it is poorly installed.

The circular movement showed similar results to the previous ones. It is clear that a line scan sensor can measure longitudinal velocity even when the movement is not linear. A automobile has a typical turning diameter of about 10 m and here was shown that the sensor can perform well to diameters as low as 2 m. The trajectories estimated were close to the ground truth, except when the longitudinal velocity is poorly estimated. The error in the trajectory is mostly due to the longitudinal velocity deviations and not due to gyroscopes deviations.

For the oscillatory movement it was shown that measuring the velocity with a small change on the optics magnification is possible, although with higher errors than the previous situations. The measurements showed some fluctuations but on average the error was small and the total distance computed had a very small error, except for stone

were the average error was 12.84 %. The surface texture is an important variable in this type of movement, as a texture-less surface like stone can induce higher errors. It might be possible to solve this problem with telecentric lenses, which have a magnification invariable with the target distance.

Looking at the tables of results it is possible to see that the value of the average error value was the same in many different situations. A value that appeared often was 1.03% for the target velocities of 500 and 1000 mm/s. This is due to the low resolution. At 500 mm/s the value of displacement calculated was 8. At 1000 mm/s the displacement calculated was exactly the double the previous, 16 pixels, and velocity is the double so the error is exactly the same.

5.2 Results with Dynamic Sub-Sampling

After the previous results it is clear that this method has high error when dealing with low velocities; to improve the measurements, the resolution of the sensor would need to be higher. Another alternative to improve the precision is to reject the lines that are close to each other. In this section is proposed an improvement to the algorithm to increase the accuracy at low velocities.

This algorithm consists in using one reference line, then, each line captured is correlated to the reference until a minimum displacement is found. After the minimum displacement is found the current line becomes the new reference. The process is illustrated in Figure 5.18. The distance can be calculated with the algorithm for the similarity coefficients or cross-correlation.

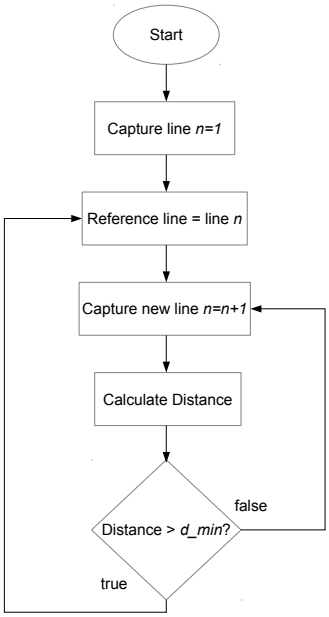


Figure 5.18: Illustration of the algorithm for dynamic sub-sampling. Where n represents the line number and d_min represents the minimum displacement.

This will reduce the sampling rate because the intermediate displacements are discarded but will improve the ability to measure low velocities. This was considered to be Dynamic Sub-Sampling.

This algorithm was applied to all the experiments, as in the previous section, and the results are shown in the following subsections. The minimum displacement required was 16 pixels.

5.2.1 Parallel Movement

the velocity estimated was highly similar for all of the materials. Figures 5.19, 5.20, 5.21, 5.22, 5.23, 5.24 show the plots for the velocity calculated with the data collected from camera. Each figure shows, also, the velocity plot from the robot (ground truth). The measurements presented were made on concrete and it should be emphasized that the plots for the other materials are similar.

Table 5.14 show the velocity errors for all the movements with all the materials. For the velocity it was calculated the average error using the robot as reference. It was also calculated the error for the total distance traveled. The results show that the tendency of the error is the same to all the materials.

With sub-sampling the big offsets are no longer present, as expected, and the velocity plots are now more similar to the ground truth plots.

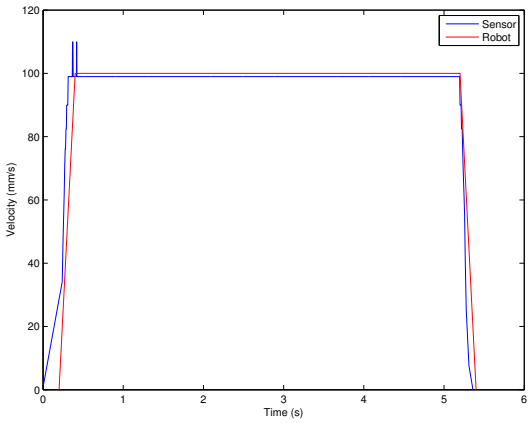


Figure 5.19: Parallel movement - Velocity plot - 100 mm/s at distance of 200 mm with dynamic sub-sampling.

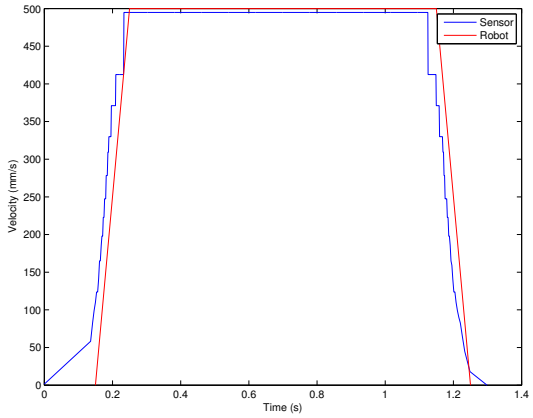


Figure 5.20: Parallel movement - Velocity plot - 500 mm/s at distance of 200 mm with dynamic sub-sampling.

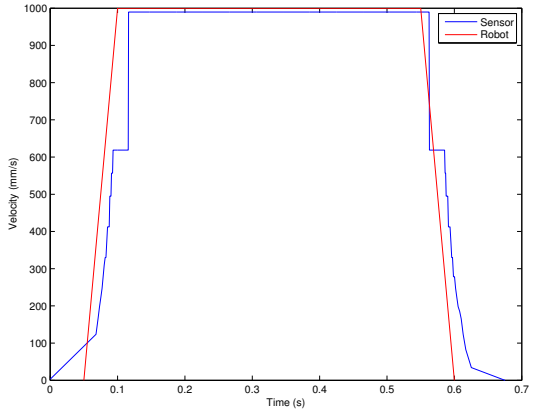


Figure 5.21: Parallel movement - Velocity plot - 1000 mm/s at distance of 200 mm with dynamic sub-sampling.

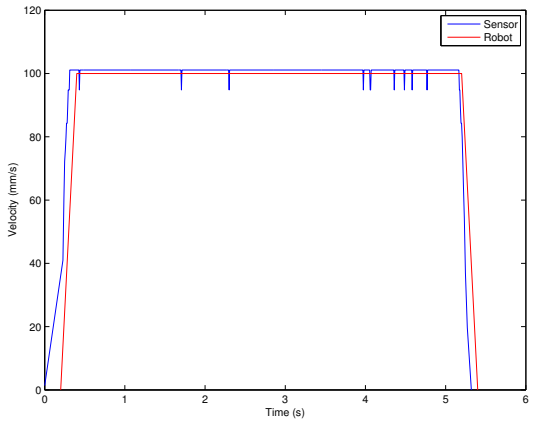


Figure 5.22: Parallel movement - Velocity plot - 100 mm/s at distance of 300 mm with dynamic sub-sampling.

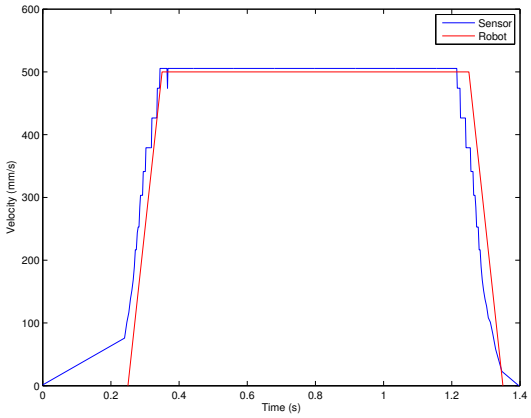


Figure 5.23: Parallel movement - Velocity plot - 500 mm/s at distance of 300 mm with dynamic sub-sampling.

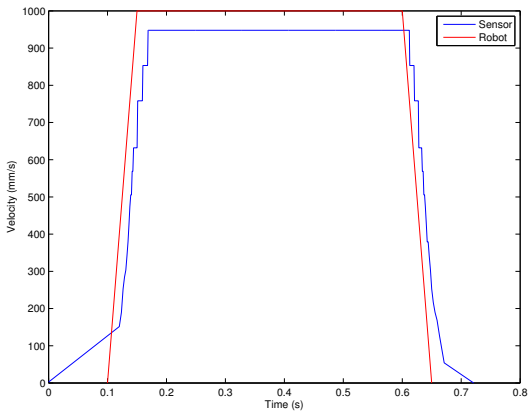


Figure 5.24: Parallel movement - Velocity plot - 1000 mm/s at distance of 300 mm with dynamic sub-sampling.

Table 5.14: Parallel movement - Summary of error results with dynamic sub-sampling.

		Concrete		Cork		Stone		Asphalt	
Separation Distance (mm)	Target Velocity (mm/s)	Avg. Vel. (%)	Dist. Travel. (%)	Avg. Vel. (%)	Dist. Travel. (%)	Avg. Vel. (%)	Dist. Travel. (%)	Avg. Vel. (%)	Dist. Travel. (%)
200	100	1.03	0.95	1.03	0.95	0.96	0.95	1.03	0.95
	500	1.03	0.99	1.03	0.96	1.03	0.98	1.03	0.99
	1000	1.03	2.41	1.03	2.30	1.03	2.35	1.03	2.23
300	100	1.09	1.07	1.03	0.94	0.99	0.96	1.07	1.03
	500	1.10	0.69	1.11	0.56	1.11	0.70	1.10	0.70
	1000	5.21	4.93	5.21	5.03	5.21	4.94	5.21	4.92

5.2.2 Diagonal Movement

For the diagonal movement the velocity plots were highly similar to the parallel one. Again, with sub-sampling there no more big offsets on the measurements.

Table 5.15 shows the average velocity and total distance relative error for all the movements with all the materials at the distance of 200 *mm*. Table 5.4 shows the results for a distance of 300 *mm*. It was calculated the average error and the distance using the robot as reference.

Here it is possible to see that the tendency of the error is the same as for the parallel movement. In fact, the results were similar in the two cases, having even equal results in most of the movements. As seen in the previous section,the method continues robust even when the image sensor has a rotation relative to the line of movement.

Table 5.15: Diagonal movement - errors for a distance separation of 200 *mm* with dynamic sub-sampling.

		Concrete		Cork		Stone		Asphalt	
Angle (°)	Target Velocity (mm/s)	Avg. Vel. (%)	Dist. Travel. (%)	Avg. Vel. (%)	Dist. Travel. (%)	Avg. Vel. (%)	Dist. Travel. (%)	Avg. Vel. (%)	Dist. Travel. (%)
2.5	100	1.02	0.79	1.02	0.79	0.96	0.71	1.02	0.79
	500	1.03	0.93	1.03	0.21	1.03	0.30	1.03	0.62
	1000	1.03	1.70	1.03	1.59	1.03	1.86	1.03	1.42
5	100	1.02	0.08	1.03	0.56	0.92	0.40	1.02	0.13
	500	1.03	0.29	1.03	0.18	1.03	0.28	1.03	0.33
	1000	1.03	1.12	1.03	1.63	1.03	1.74	1.03	1.24
10	100	0.97	3.83	0.37	1.19	0.81	0.32	0.86	1.10
	500	0.99	0.46	1.00	0.44	1.01	0.63	1.00	0.75
	1000	1.07	2.01	1.07	1.34	1.03	1.07	1.07	1.95

Table 5.16: Diagonal movement - errors for a distance separation of 300 *mm* with dynamic sub-sampling.

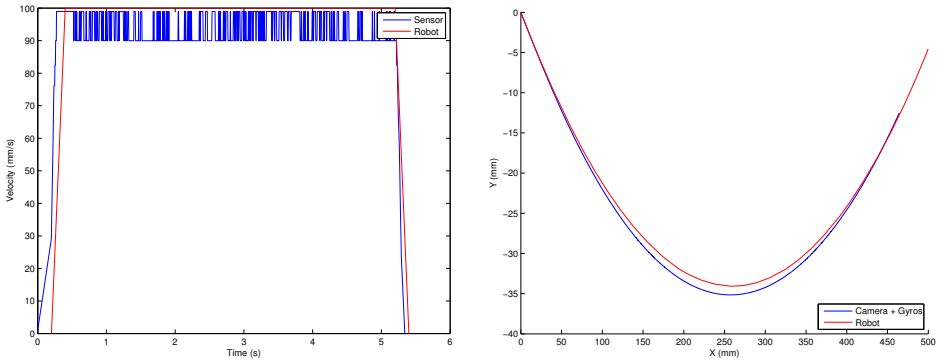
		Concrete		Cork		Stone		Asphalt	
Angle (°)	Target Velocity (mm/s)	Avg. Vel. (%)	Dist. Travel. (%)	Avg. Vel. (%)	Dist. Travel. (%)	Avg. Vel. (%)	Dist. Travel. (%)	Avg. Vel. (%)	Dist. Travel. (%)
2.5	100	1.09	1.19	1.01	1.07	1.09	1.02	1.09	1.12
	500	1.10	1.40	1.11	1.14	1.11	1.31	1.10	1.39
	1000	5.21	4.20	5.21	4.29	5.21	4.21	5.21	4.23
5	100	0.64	1.43	0.97	1.31	1.10	1.55	0.86	1.21
	500	0.16	0.69	0.93	1.02	1.10	1.41	0.65	0.70
	1000	5.21	3.64	5.21	4.23	5.21	4.20	5.21	3.72
10	100	0.28	1.19	0.40	1.07	0.94	1.91	0.31	1.02
	500	0.97	0.10	1.23	0.44	0.33	1.23	0.99	0.24
	1000	5.26	3.76	5.21	3.55	5.21	3.58	5.21	3.40

5.2.3 Circular Movement

Figures 5.25, 5.26, 5.27, 5.28, 5.29, 5.30 show the generic velocity plots for the circular movement at the various velocities and distances. It shows also the trajectory made by the robot versus the one estimated by the camera and gyroscopes.

Tables 5.17, 5.18, 5.19, 5.20, 5.21, 5.22, 5.23, 5.24 show the error and final position deviation for all the experiments. The tables show the average velocity error and total distance error, they also show the coordinates deviation and the euclidean distance between the final position estimated and the ground truth.

An improvement using dynamic sub-sampling is clear, mostly for the plots at 100 mm/s because of the attenuation of the offset seen in the previous section.



(a) (b)

Figure 5.25: Velocity and position plot for velocity of 100 mm/s - diameter of 2 m - at distance 200 mm with dynamic sub-sampling.

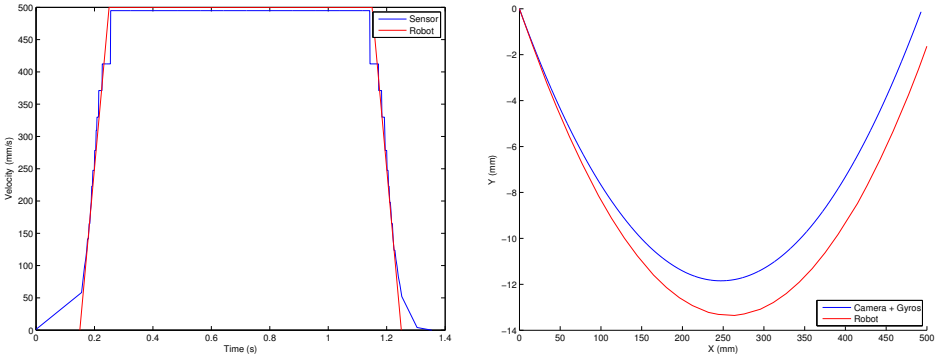


Figure 5.26: Velocity and position plot for velocity of 500 mm/s - diameter of 5 m - at distance 200 mm with dynamic sub-sampling.

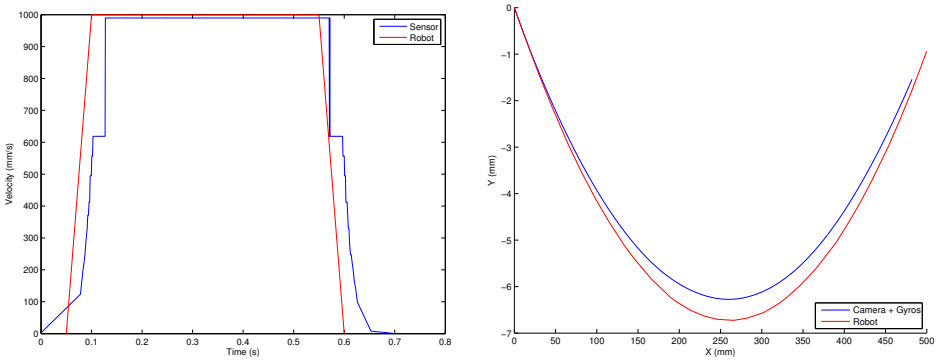


Figure 5.27: Velocity and position plot for velocity of 1000 mm/s - diameter of 10 m - at distance 200 mm with dynamic sub-sampling.

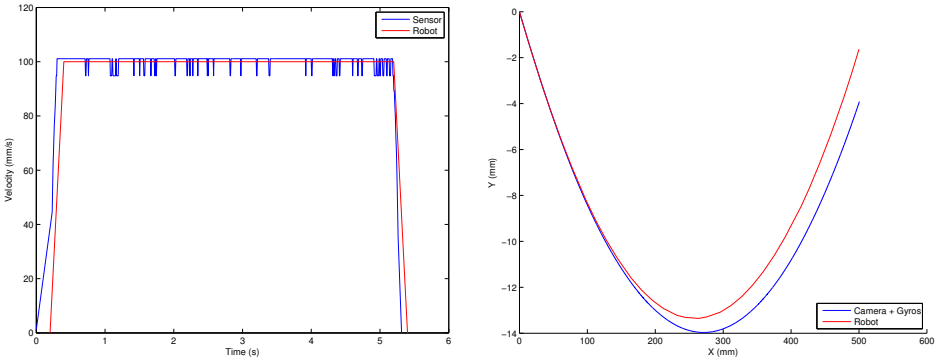


Figure 5.28: Velocity and position plot for velocity of 100 mm/s - diameter of 5 m - at distance 300 mm with dynamic sub-sampling.

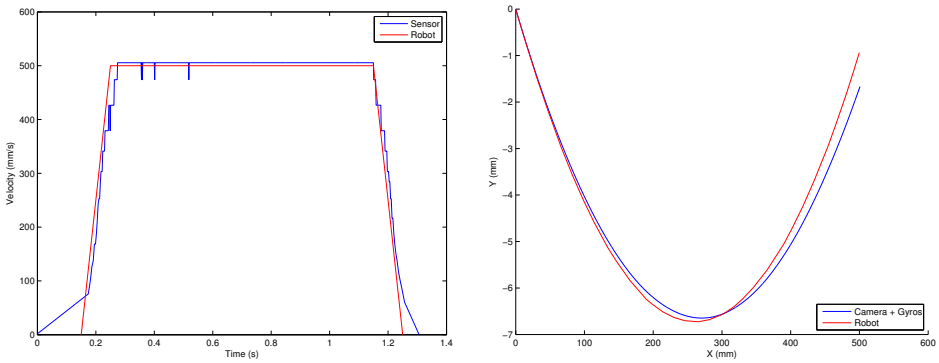


Figure 5.29: Velocity and position plot for velocity of 500 mm/s - diameter of 10 m - at distance 300 mm with dynamic sub-sampling.

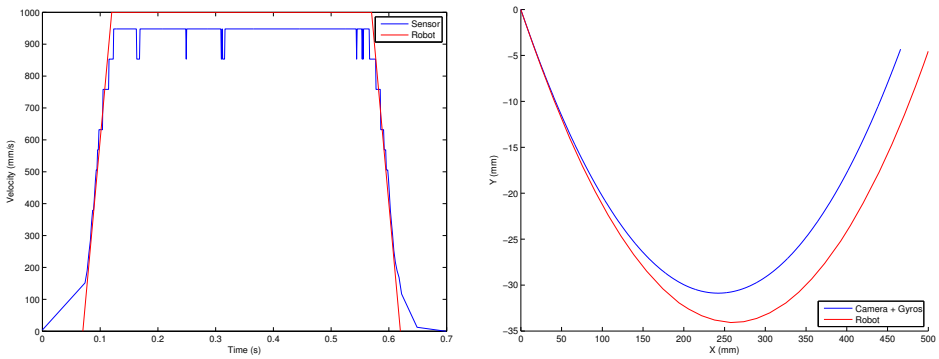


Figure 5.30: Velocity and position plot for velocity of 1000 mm/s - diameter of 2 m - at distance 300 mm with dynamic sub-sampling.

Table 5.17: Circular movement - Errors for concrete at separation distance of 200 mm using dynamic sub-sampling.

			Concrete				
			Error		Deviation		
Separation Distance (mm)	Diameter (mm)	Target Velocity (mm/s)	Avg. Vel. (%)	Dist. Travel. (%)	X (mm)	Y (mm)	Euclidean Distance (mm)
200	2	100	6.13	5.31	35.56	7.97	36.45
		500	13.71	12.58	66.02	1.90	66.05
		1000	27.63	29.53	156.35	0.53	156.35
	5	100	1.03	2.57	15.10	1.13	15.14
		500	1.03	0.71	7.20	1.50	7.36
		1000	1.03	2.28	17.86	1.22	17.90
	10	100	1.03	1.03	5.69	2.18	6.09
		500	1.03	1.25	8.34	0.55	8.36
		1000	1.03	2.68	17.70	0.61	17.71

Table 5.18: Circular movement - Errors for concrete at separation distance of 300 mm using dynamic sub-sampling.

			Concrete				
			Error		Deviation		
Separation Distance (mm)	Diameter (mm)	Target Velocity (mm/s)	Avg. Vel. (%)	Dist. Travel. (%)	X (mm)	Y (mm)	Euclidean Distance (mm)
300	2	100	5.21	4.27	27.13	7.19	28.07
		500	5.43	4.41	30.81	0.68	30.81
		1000	5.38	4.53	33.87	0.23	33.87
	5	100	0.55	0.58	0.62	2.28	2.36
		500	0.41	0.31	1.61	1.24	2.03
		1000	5.21	4.52	28.22	1.02	28.24
	10	100	1.11	1.07	4.37	2.31	4.95
		500	1.07	0.61	1.03	0.73	1.26
		1000	5.21	4.93	28.32	0.72	28.33

Table 5.19: Circular movement - Errors for cork at separation distance of 200 mm using dynamic sub-sampling.

			Cork				
			Error		Deviation		
Separation Distance (mm)	Diameter (mm)	Target Velocity (mm/s)	Avg. Vel. (%)	Dist. Travel. (%)	X (mm)	Y (mm)	Euclidean Distance (mm)
200	2	100	4.94	4.12	30.11	8.76	31.36
		500	11.65	11.33	57.03	1.89	57.06
		1000	23.50	27.19	146.83	2.67	146.85
	5	100	1.03	0.71	5.01	1.96	5.37
		500	1.03	0.61	7.29	1.70	7.48
		1000	1.03	2.03	17.33	1.44	17.39
	10	100	1.03	1.03	5.79	2.17	6.19
		500	1.03	1.20	9.07	0.50	9.09
		1000	1.03	2.64	17.77	0.52	17.77

Table 5.20: Circular movement - Errors for cork at separation distance of 300 mm using dynamic sub-sampling.

			Cork				
			Error		Deviation		
Separation Distance (mm)	Diameter (mm)	Target Velocity (mm/s)	Avg. Vel. (%)	Dist. Travel. (%)	X (mm)	Y (mm)	Euclidean Distance (mm)
300	2	100	5.32	4.27	27.09	5.99	27.74
		500	8.63	7.58	46.48	0.27	46.48
		1000	8.71	7.87	50.38	0.27	50.38
	5	100	0.70	0.51	5.33	2.29	5.80
		500	1.64	2.02	13.40	1.22	13.45
		1000	5.21	4.51	28.92	1.16	28.95
	10	100	0.62	0.45	0.92	2.29	2.46
		500	0.91	0.03	1.57	0.51	1.65
		1000	5.21	5.18	30.41	0.60	30.42

Table 5.21: Circular movement - Errors for stone at separation distance of 200 *mm* using dynamic sub-sampling.

			Stone				
			Error		Deviation		
Separation Distance (*mm*)	Diameter (*mm*)	Target Velocity (mm/s)	Avg. Vel. (%)	Dist. Travel. (%)	X (*mm*)	Y (*mm*)	Euclidean Distance (*mm*)
200	2	100	2.60	1.66	17.34	8.34	19.24
		500	5.46	5.47	24.70	2.79	24.86
		1000	10.86	15.76	94.32	5.01	94.45
	5	100	1.03	0.71	4.96	1.96	5.33
		500	1.03	0.52	6.65	1.65	6.85
		1000	1.03	1.92	16.48	1.43	16.54
	10	100	1.03	1.030	5.70	2.18	6.11
		500	1.03	1.13	7.81	0.41	7.82
		1000	1.03	2.70	17.56	0.47	17.57

Table 5.22: Circular movement - Errors for stone at separation distance of 300 *mm* using dynamic sub-sampling.

			Stone				
			Error		Deviation		
Separation Distance (*mm*)	Diameter (*mm*)	Target Velocity (mm/s)	Avg. Vel. (%)	Dist. Travel. (%)	X (*mm*)	Y (*mm*)	Euclidean Distance (*mm*)
300	2	100	5.26	4.27	27.19	7.12	28.11
		500	8.20	7.12	44.37	0.45	44.37
		1000	7.87	7.13	46.80	0.16	46.80
	5	100	0.77	0.63	5.91	2.34	6.36
		500	2.30	2.33	14.94	1.20	14.99
		1000	5.21	4.54	28.98	1.10	29.00
	10	100	0.65	0.45	0.85	2.29	2.44
		500	0.89	0.01	1.32	0.51	1.42
		1000	5.21	5.18	30.15	0.60	30.16

Table 5.23: Circular movement - Errors for asphalt at separation distance of 200 *mm* using dynamic sub-sampling.

			Asphalt				
			Error		Deviation		
Separation Distance (*mm*)	Diameter (*mm*)	Target Velocity (mm/s)	Avg. Vel. (%)	Dist. Travel. (%)	X (*mm*)	Y (*mm*)	Euclidean Distance (*mm*)
200	2	100	6.13	5.30	35.56	7.97	36.44
		500	13.70	12.58	66.01	1.89	66.04
		1000	27.62	29.53	156.34	0.52	156.34
	5	100	1.03	2.57	15.10	1.13	15.14
		500	1.03	0.71	7.20	1.49	7.35
		1000	1.03	2.27	17.85	1.22	17.89
	10	100	1.030	1.03	5.68	2.18	6.09
		500	1.03	1.24	8.34	0.55	8.36
		1000	1.03	2.68	17.69	0.60	17.70

Table 5.24: Circular movement - Errors for asphalt at separation distance of 300 *mm* using dynamic sub-sampling.

			Asphalt				
			Error		Deviation		
Separation Distance (*mm*)	Diameter (*mm*)	Target Velocity (mm/s)	Avg. Vel. (%)	Dist. Travel. (%)	X (*mm*)	Y (*mm*)	Euclidean Distance (*mm*)
300	2	100	5.21	4.27	27.13	7.19	28.06
		500	5.42	4.40	30.80	0.67	30.81
		1000	5.38	4.53	33.86	0.23	33.86
	5	100	0.55	0.58	0.61	2.28	2.36
		500	0.40	0.30	1.61	1.23	2.03
		1000	5.21	4.51	28.22	1.02	28.24
	10	100	1.10	1.06	4.37	2.30	4.94
		500	1.06	0.61	1.02	0.73	1.26
		1000	5.21	4.92	28.32	0.72	28.33

5.2.4 Oscillatory Movement

The following figures show the velocity plots for each material. Figure 5.31 shows the velocity plot for concrete, Figure 5.32 shows the result for cork and Figure 5.33 shows the result for stone.

Table 5.25 shows the errors for these experiments. The average and maximum error were calculated for velocity and the error for total distance traveled was also calculated.

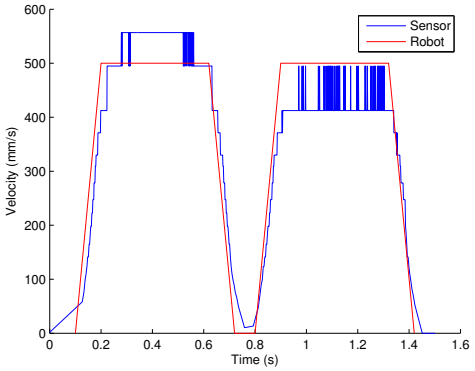


Figure 5.31: Oscillatory movement - Velocity plot - 500 mm/s at distance of 200 mm - concrete with dynamic sub-sampling.

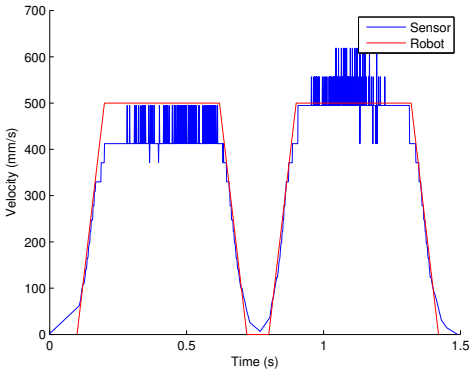


Figure 5.32: Oscillatory movement - Velocity plot - 100 mm/s at distance of 200 mm - Cork with dynamic sub-sampling.

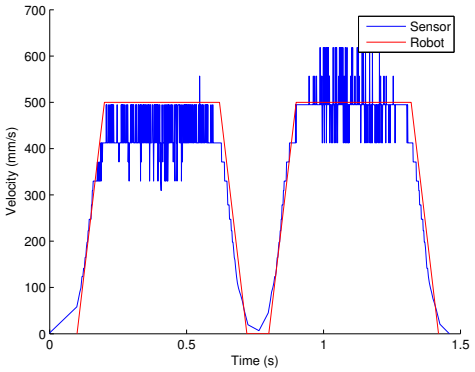


Figure 5.33: Oscillatory movement - Velocity plot - 100 mm/s at distance of 200 mm - stone with dynamic sub-sampling.

Table 5.25: Error Results for oscillatory movement using dynamic sub-sampling.

	Velocity (mm/s)	Avg. Velocity Error (%)	Total Distance Error (%)
Concrete	500	2.65	1.26
Cork		4.96	2.77
Stone		5.69	3.70
Asphalt		2.55	1.16
Concrete	1000	2.45	1.36
Cork		4.91	2.87
Stone		5.53	3.10
Asphalt		2.65	1.26

5.2.5 Results Discussion

On the parallel movement the use of dynamic sub-sampling improved the results mostly on the experiments at 100 mm/s. On the other situations the average error was the same or had small variations. The velocity plots now have an acceleration phase that fits better with the ones from the ground truth opposed to the degree-like acceleration seen on the previous section.

The results were consistent on the diagonal movements and the same tendency seen previously occurred here too. The error results are very similar to the ones on the parallel movement showing that the angle of the image sensor had approximately none impact.

For circular movements the improvements made by this algorithm were similar to the previous situations. For 100 mm/s the error was significantly reduced, in some cases from around 23% to around 5%. Although with a diameter of 2 m the estimates for 500 and 1000 mm/s had an increased error. This is due to the fact that 2 meters of diameter is a very tight curvature and that produces high rotation between the successive line scans and thus a weaker similarity; then this algorithm reduces the sampling rate by discarding some line scans and the correlation is made with more distant lines wish have even weaker similarity. The similarity between lines is, of course, an essential parameter and as the similarity decreases the error increases. Overall, for the other cases, the errors maintained very similar to results with the previous algorithm. For tighter curvatures the correlation only works with good results if sub-sampling is not used. For this reason the algorithm may dynamically changes its minimum distance of pixel displacement based on the data from the gyroscopes, to not compromise the sampling rate. Basically as the curvature increases this minimum needs to be decreased. This solution would change its level of sub-sampling to better suit the needs.

For the oscillatory movement the results were similar to the ones using the previous algorithm. Only there was an improvement for the experiments with stone. In this case, with the previous algorithm and as stated in Figure 5.16, the velocity measured was in some cases zero; with the sub-sampling it was imposed a minimum displacement to be read and that eliminated those bad estimates.

Looking at the tables of results it is possible to see that the value of the average error value was the same in many different situations. A value that appeared often was 1.03% for the target velocities of 500 and 1000 mm/s. This is due to the low resolution. At 500 mm/s the value of displacement calculated was 8. At 1000 mm/s the displacement

calculated was exactly the double the previous, 16 pixels, and velocity is the double so the error is exactly the same.

This algorithm using sub-sampling reveals clear improvements when dealing with low velocities without compromise for higher velocities. Although for it to operate with guarantees it should adjust its "level" of sub-sampling based on how much the vehicle is turning.

5.3 Parallel Movement With Gyroscopes

It is important to study the influence of the gyroscopes readings and its deviation to the movement estimate when moving straight forward. In this case, due to the gyroscope's drift, an error of orientation will appear. To study this effect it was used the previous velocity plots from the parallel movement together with the gyroscopes readings. The results are shown in Figures 5.34, 5.35, 5.36, 5.37, 5.38, 5.39. Table 5.26 shows the final positions of each simulation using the longitudinal velocity estimated by the camera and the gyroscope's readings. The reference was calculated using the same longitudinal velocity estimations but with a reference angular velocity of 0 degrees per second.

On table 5.26 it is possible to see that the deviation is small and the higher the longitudinal velocity error the higher the final position deviation. Including the gyroscopes data in this simulation had very little impact on the movement estimation. This proves that even when the movement is a straight line, the gyroscope's drift does not degrade the position estimates. However, these deviations can be significant as the distance increases, then higher precision, low drift gyroscopes may be used.

In Figure 5.37 it is possible to see that the velocity plot has high oscillations. In this case the displacements calculated were 0 or 2 pixels, equivalent to a velocity of 0 or 189.5735 mm/s. This happened because the successive lines were so similar and the resolution was low causing the correlation to fail in some cases.

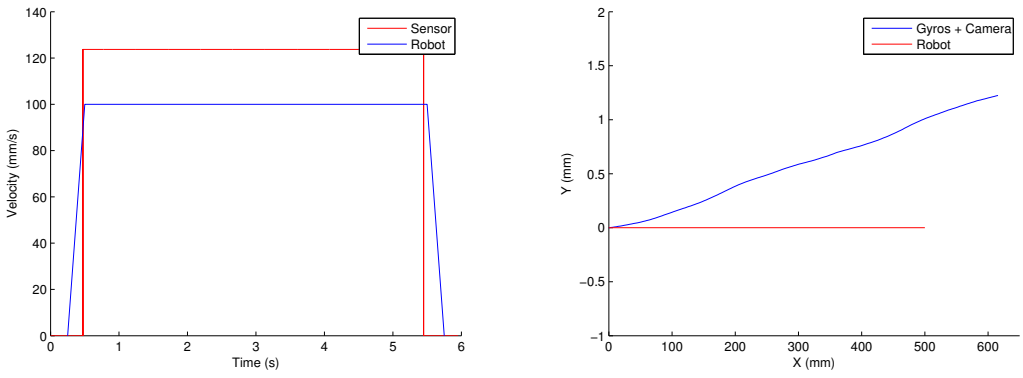


Figure 5.34: Velocity and position plot for velocity of 100 mm/s at distance 200 mm.

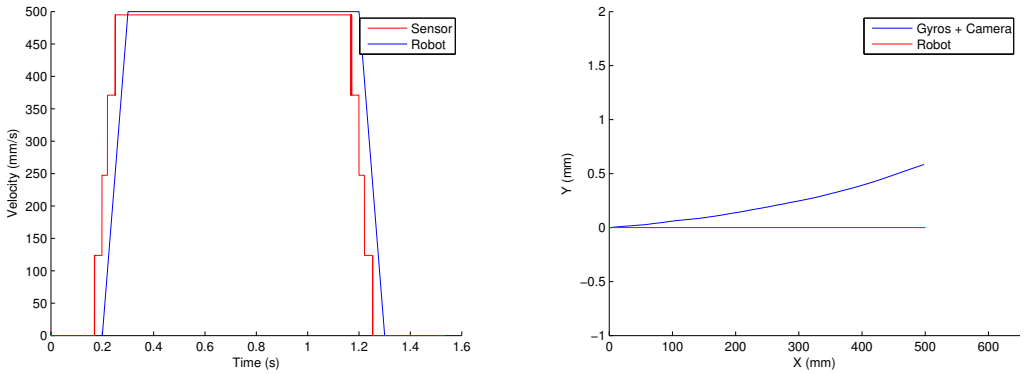


Figure 5.35: Velocity and position plot for velocity of 500 mm/s at distance 200 mm.

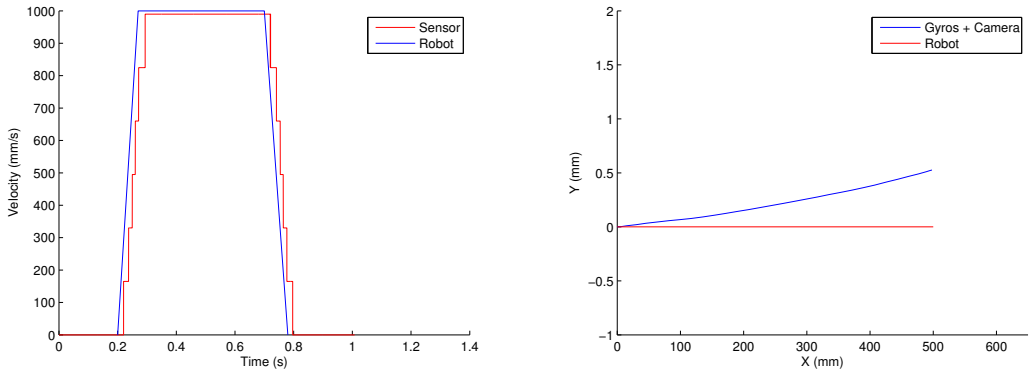


Figure 5.36: Velocity and position plot for velocity of 1000 mm/s at distance 200 mm.

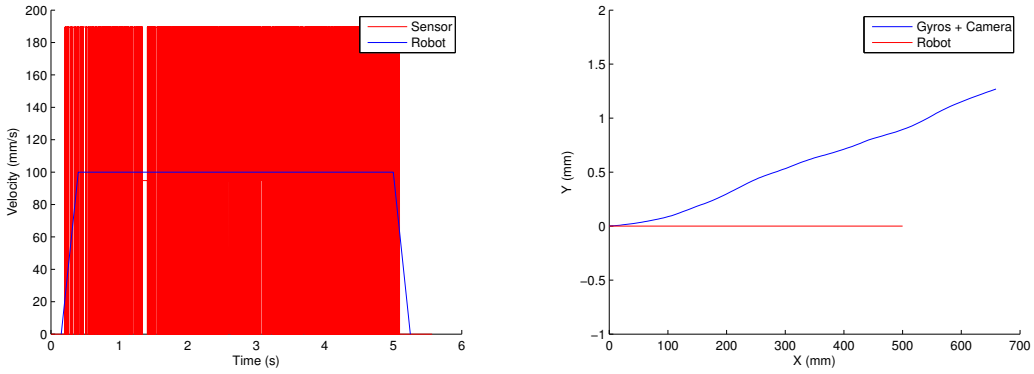


Figure 5.37: Velocity and position plot for velocity of 100 mm/s at distance 300 mm.

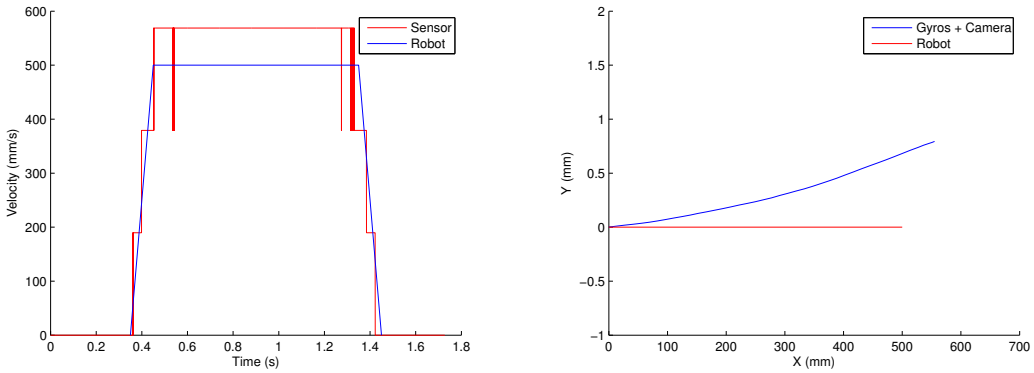


Figure 5.38: Velocity and position plot for velocity of 500 mm/s at distance 300 mm.

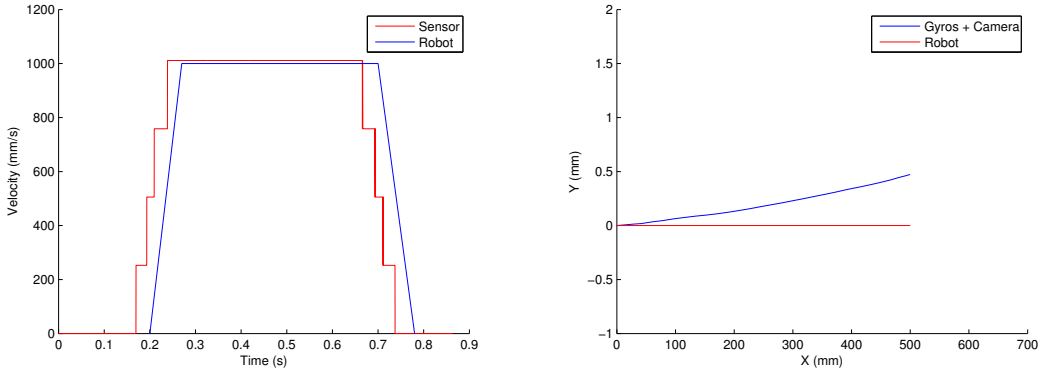


Figure 5.39: Velocity and position plot for velocity of 1000 mm/s at distance 300 mm.

Table 5.26: Final positions for parallel movement with gyroscope's data.

Distance	Velocity (mm/s)	X reference (mm)	X estimated (mm)	Y reference (mm)	Y estimated (mm)
200	500	621.00	615.78	0.00	1.22
		497.50	498.06	0.00	0.58
		496.20	498.66	0.00	0.53
300	1000	704.30	659.15	0.00	1.27
		558.25	555.07	0.00	0.79
		497.66	499.56	0.00	0.47

5.4 Correlation with Interpolation

One limitation of the algorithms used is that they only estimate a pixel displacement that is an integer. For low velocities this resolution might be small and that can induce high errors. A solution that can reduce the error is by doing the correlation with interpolation to estimate displacements that are floats.

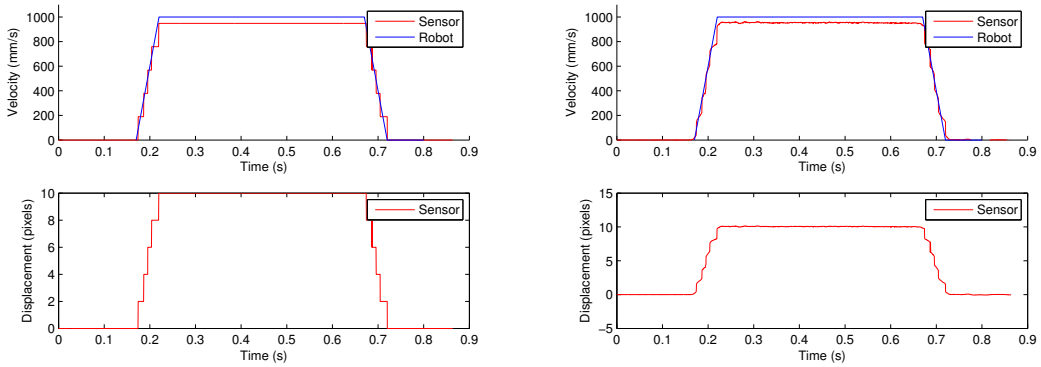
The interpolation is a quadratic interpolation of 3 uniformly spaced samples. After applying one of the algorithms there will be a correlation coefficient for each equally spaced shift. This method performs a quadratic interpolation between the maximum correlation coefficient and its neighbors.

In this section there are presented four examples with the application of this improvement. Figure 5.40, 5.41, 5.42, 5.43 show the velocity and pixel displacement plots of the original algorithm (a) versus the algorithm with interpolation (b).

Table 5.27: Error results for Normal Correlation Vs Correlation with Interpolation.

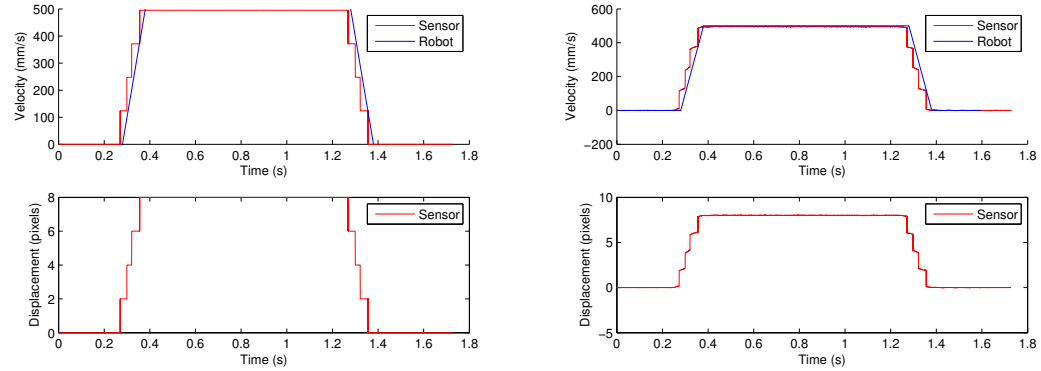
	Without Interpolation		With Interpolation	
	Avg. Velocity Error (%)	Distance Error (%)	Avg. Velocity Error (%)	Distance Error (%)
Example 1	5.12	5.12	4.55	4.63
Example 2	1.03	0.92	0.89	0.88
Example 3	6.17	4.48	3.20	2.42
Example 4	1.14	0.95	1.13	0.94

Analyzing table 5.27 it is possible to see that using interpolation approximates the velocity estimates to the ground truth and the average error decreases. Interpolation can then be successfully used to increase the accuracy of the estimates with no compromise in computation speed, since interpolation is a simple operation.



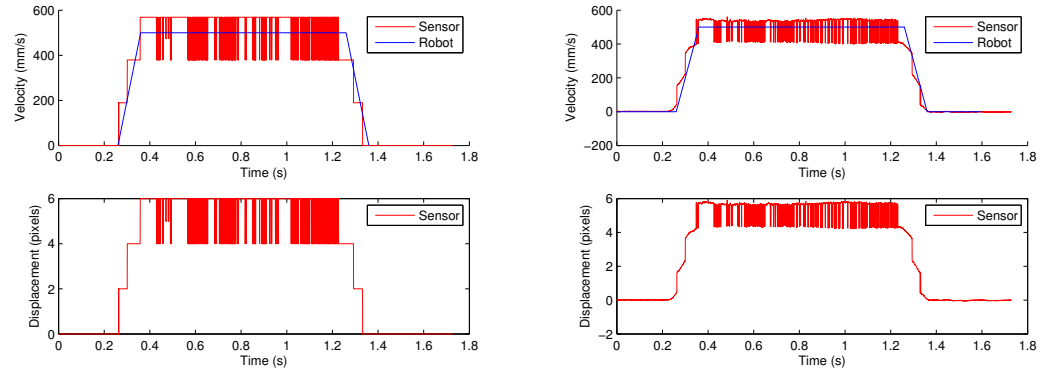
(a) Whitout interpolation (b) With interpolation

Figure 5.40: Example 1 - 1000 mm/s at distance of 300 mm.



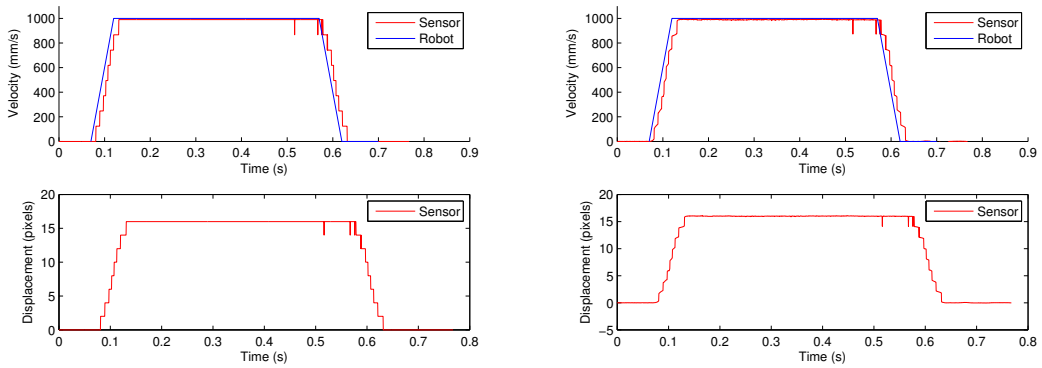
(a) Without interpolation (b) With interpolation

Figure 5.41: Example 2 - 500 mm/s at distance of 200 mm.



(a) Without interpolation (b) With interpolation

Figure 5.42: Example 3 - 500 mm/s at distance of 300 mm.



(a) Without interpolation (b) With interpolation

Figure 5.43: Example 4 - 1000 mm/s at distance of 200 mm.

5.5 Sub-sampling

Sub-sampling is the process of reducing the sampling rate. When performing a sub-sample of the video data, one can simulate a higher displacement and thus a higher velocity movement. For example, using the 100 mm/s video and doing the computing on intervals of 5 frames, the displacement will be 5 times higher and thus the velocity, also 5 times higher. With sub-sampling it is possible to make a extrapolation of the data to higher velocities.

To prove that using sub-sampling gives the same results as using the full sample some simulations were made. A sample from 100 mm/s was used to simulate a movement with a velocity of 500 and 1000 mm/s the errors were exactly the same as the ones calculated using the full sample. The same was done with a sample form 500 mm/s simulating a movement of 1000 mm/s and the error was also equal. This leads to the conclusion that it is safe to do sub-sampling and extrapolate to higher velocities.

Next, simulations of 5, 10 and 20 m/s are made with the sample taken at 100 mm/s on concrete. The results are shown in Figure 5.44, 5.45 and 5.46 and the error results are presented on Table 5.28. These results show that at a sampling rate of 2500 Hz the sensor can measure velocity up to 10 m/s with an average error of 0.2 % and a maximum error of 5.15 %. At 20 m/s the results begin to show relevant fluctuations and a maximum error of 35.67 % is reached.

Table 5.28: Error results for sub-sampling simulations.

Velocity (m/s)	Avg. Error (%)	Max. Error (%)
5	0.2	3.918
10	0.18	5.15
20	1.05	35.67

The results of the sub-sampling show that increasing the displacement increases the global and the average error. At an acquisition frequency of 2500 Hz the measurements are acceptable up to 10 m/s, if dealing with higher velocities one should increase the sampling rate. This leads to the conclusion that for car-like velocities it is necessary to

increase the sampling rate. For example if one wants to measure a velocity up to 120 km/h or 33.333 m/s, it would be necessary a sampling rate of $3.3333 \times 2500 = 8333.33$ Hz.

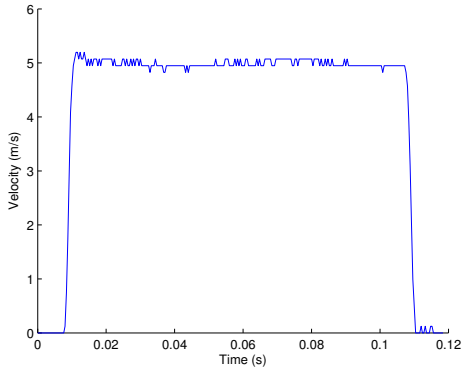


Figure 5.44: Velocity plot at 5 m/s using sub-sampling from sample of 100 mm/s.

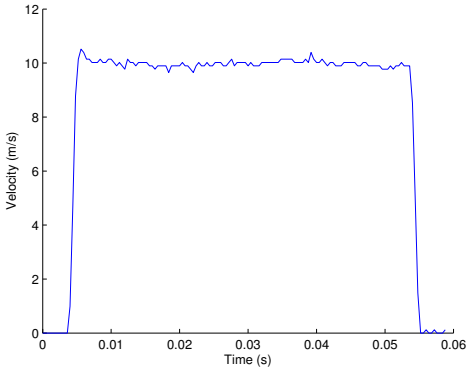


Figure 5.45: Velocity plot at 10 m/s using sub-sampling from sample of 100 mm/s.

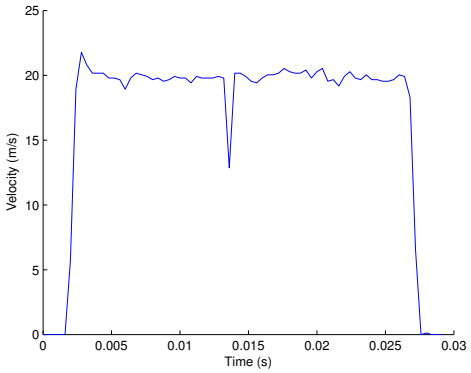


Figure 5.46: Velocity plot at 20 m/s using sub-sampling from sample of 100 mm/s.

Chapter 6

Conclusions and Future Work

6.1 Conclusions

To meet the proposed objectives, a study of the state of the art on odometry solutions was developed. After that, a study was made to find the most suitable solution for the problem and due to the advantages of flexibility, a visual odometry with a line scan image sensor was the solution chosen for study.

Visual odometry with line scan sensors is a new area of study and very little literature exists. So the main objective of this wotk was to find out the problems, limitations and the relevant variables for the operation of this method.

It was proven that a line scan image sensor can operate as a velocity sensor under controlled conditions for lighting and absence of vibration. Under these conditions the sensor can estimate velocity with low error, about 1% on most of the cases. The sensor performed well for straight line movements with the camera aligned and misaligned up to a maximum of 10°. For circular movements the sensor performed well and showed that it is possible to operate when an automobile is executing maneuvers with a tight turning radius. For situations when the focal distance is changing the results were satisfactory for concrete and cork although they were poor for stone. This is a limitation for the hardware used and can be improved using appropriate optics.

This method creates new possibilities for visual odometry. The most limiting features of common visual odometry are the low velocity that can be estimated, due to low frame rates of the common cameras; the complexity of the algorithms which are needed to be used causing high computational cost and difficulty of real time application. The line scan sensor needs less complex algorithms and can easily achieve high frame rates necessary to measure high velocities.

The method shows promising results and further investigation and development should be performed in order to study the applicability in real world scenarios.

6.2 Future Work

The results from this work showed that using a line scan sensor to measure velocity of a automobile is possible in controlled conditions. The results are promising and further investigation should be developed. Several variables should be investigated, like the effect of lighting variation; the effect of introducing vibration and the effect of changing the

focal distance. Outdoors experiments should also be realized with all those variables introduced in the environment.

The optics used performed acceptably with variable focal length. Although to improve even more the consistency of the measurements telecentric lens may be used, this lens have a magnification that is invariable with the focal distance. This means that even when with irregular ground or with oscillations from the vehicle suspension, the estimates will be more precise. Other solution for high focal distance variations might be a combination of several lenses, each calibrated to operate at a specific distance. Then, the equipment along with the distance information from another sensor, like a laser sensor, can commute between the lenses to use the one most appropriated for the current distance.

To ensure that this method has good performance when dealing with circular movements, the image sensor should be as tangential as possible to the line of movement. When the image sensor is not tangent that means that the line scans will have a lateral displacement; an high lateral displacement will eventually cause the system to fail computing the correct longitudinal displacement. To ensure that the image sensor is always tangent to the line of movement, a system using a servo motor relaying on the gyroscopes readings can be used to align the image sensor. Basically the image sensor would be mounted on a servo motor that would change its orientation as much as the rotation given by the gyroscopes.

The use of an image sensor with higher sensitivity should be considered. In this work the acquisition rate was limited to 2500 Hz due to the lack of brightness on the images. This was caused by the low exposure time, causing the images to be dark. To solve this, a higher sensitivity image sensor should be used to not compromise the quality of the images at high frame rates.

In a real world scenario infra-red lighting can be used, as this type of light is invisible to the human eye. This is important if the sensor is to be used by automobiles on the public roads; it is important that the lighting for the sensor isn't a distraction for the drivers. Also, the artificial infra-red lighting should be able overcome the infra-red radiation from the sun. This is important as the radiation from the sun varies throughout the day and the lighting should be as invariable as possible. Then the image sensor should be sensitive to the wavelength of the infra-red radiation.

References

[1] Universidade de Aveiro, "Atlas project." http://atlas.web.ua.pt/, May 2014.

[2] V. Santos, J. Almeida, and E. Ávila, "ATLASCAR-technologies for a computer assisted driving system on board a common automobile," *Annual Conference on Intelligent Transportation Systems*, pp. 1421–1427, 2010.

[3] T. Rocha, "Piloto automático para controlo e manobras de navegação do AtlasCar," Master's thesis, Universidade de Aveiro, 2011.

[4] A. V. Topalov, "Recent advances in mobile robotics," *InTech*, 2011.

[5] A. Sleeswyk, "Vitruvius' odometer," *Scientific American*, pp. 188–200, 1981.

[6] "Schickard versus pascal - an empty debate?." http://metastudies.net/pmwiki/pmwiki.php?n=Site.SchicardvsPascal, May 2014.

[7] "History of andrew p. shumway." http://www.mendonutah.net/history/personal_histories/shumway_andrew_p.htm, May 2014.

[8] Journal Sentinel, "First state flight to be commemorated in beloit." http://www.jsonline.com/news/wisconsin/69028117.html, May 2014.

[9] I. J. Cox, "An Experiment in Guidance and Navigation of an Autonomous Mobile Robot," *IEEE Transactions Robotics and Automation*, pp. 193–204, 1991.

[10] J. Borenstein, H. Everett, and L. Feng, *Where am I? Sensors and methods for mobile robot positioning*. PhD thesis, University of Michigan, 1996.

[11] J. Borenstein, H. R. Everett, L. Feng, and D. Wehe, "Mobile robot positioning: Sensors and techniques," *Journal of Robotic Systems*, vol. 14, pp. 231–249, Apr. 1997.

[12] A. Agent, "The advantages of absolute encoders for motion control," *Sensors*, pp. 19–24, 1991.

[13] J. G. Webster, *Instrumentation and Sensors Handbook*. CRC Press, 1st ed., 1998.

[14] J. Bernstein, "An overview of mems inertial sensing technology." http://www.sensorsmag.com/sensors/acceleration-vibration/an-overview-mems-inertial-sensing-technology-970, May 2014.

[15] Pololu Robotics and Electronics, "Minimu-9 v2 gyro, accelerometer, and compass." http://www.pololu.com/product/2468, May 2014.

[16] L. M. Y. Cheng, M. W. Maimone, "Visual Odometry on the Mars Exploration Rovers," *IEEE Robotics & Automation Magazine*, pp. 54–62, 2006.

[17] N. Nourani-Vatani, J. Roberts, and M. Srinivasan, "IMU aided 3D visual odometry for car-like vehicles," *Australian Robotics and Automation Association*, 2008.

[18] V. Kálmán, *On modeling and control of omnidirectional wheels.* PhD thesis, Budapest University of Technology and Economics, 2013.

[19] N. Nourani-Vatani, J. Roberts, and M. Srinivasan, "Practical visual odometry for car-like vehicles," *2009 IEEE International Conference on Robotics and Automation*, pp. 3551–3557, May 2009.

[20] Kistler, "Wheel incremental transducer wpt for acquisition of wheel rotation." `http://www.kistler.com/pt/en/product/angular-rate/CWPTA411`, May 2014.

[21] Kistler, "Kistler optical sensors." `http://www.corrsys-datron.com/optical_sensors.htm`, May 2014.

[22] Applanix Corp, "Pos lv." `http://www.applanix.com/products/land/pos-lv.html`, May 2014.

[23] J. Horn, A. Bachmann, and T. Dang, "A Fusion Approach for Image-Based Measurement of Speed Over Ground," *2006 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, pp. 261–266, Sept. 2006.

[24] N. Nourani-Vatani and P. V. K. Borges, "Correlation-based visual odometry for ground vehicles," *Journal of Field Robotics*, vol. 28, pp. 742–768, Sept. 2011.

[25] `http://www.panavision.com/`, May 2014.

[26] Hamamatsu Photonics. `http://www.hamamatsu.com/us/en/product/category/3100/4005/4120/S3901-256Q/index.html`, May 2014.

[27] Dynamax Imaging, "line scan image sensors." `http://www.dynamax-imaging.com/`, May 2014.

[28] T. F. J. Rafeiro, "Rede de Sensores Inerciais para Equilíbrio de um Robô Humanóide," Master's thesis, Universidade de Aveiro, 2013.

[29] Opto Engineering, "Telecentric lenses: Basic information and working principles." `http://www.opto-engineering.com/`, May 2014.

[30] M. Watanabe and S. Nayar, "Telecentric optics for computational vision," *Proc. of European Conference on Computer Vision*, pp. 439–451, 1995.

[31] Teledyne Dalsa, "Ccd vs. cmos." `https://www.teledynedalsa.com/imaging/knowledge-center/appnotes/ccd-vs-cmos/`, 2014.

[32] ADLINK Technolgy, "Pc-based line-scan imaging systems." `http://www.adlinktech.com/industrial_automation/PC-Based_Line-Scan.php?source=`, 2014.

[33] P. Simionescu and D. Beale, "Optimum synthesis of the four-bar function generator in its symmetric embodiment: the Ackermann steering linkage," *Mechanism and Machine Theory*, vol. 37, pp. 1487–1504, Dec. 2002.

[34] W. F. Milliken and D. L. Milliken, "Race car vehicle dynamics," *SAE*, pp. 19–24, 1995.

[35] S. Mitra and Y. Kuo, *Digital signal processing: a computer-based approach*. McGraw-Hill, 3th ed., 2006.

[36] J. P. Lewis, "Fast Template Matching Template Matching by Cross Correlation 2 Normalized Cross Correlation Normalized Cross Correlation in the Transform Domain," *Pattern Recognition*, vol. 10, no. 11, pp. 120–123, 1995.

[37] R. D. Keane and R. J. Adrian, "Theory of cross-correlation analysis of PIV images RICHARD D. KEANE & RONALD J. ADRIAN," *Kluwer Academic Publishers*, pp. 191–215, 1992.

[38] Teledyne DALSA Inc. `http://www.teledynedalsa.com/imaging/products/cameras/line-scan/piranha2/P2-4X-04K40%20(7%20uM)/`, May 2014.

[39] Rodenstock Photo Optics, "Rodagon." `http://www.rodenstock-photo.com/en/main/products/lenses-for-enlarging-and-ccd/rodagon/`, May 2014.

[40] RobotWorx, "Fanuc m-6ib/6s." `http://www.robots.com/fanuc/m-6ib-6s`, May 2014.

[41] Z. Zhao, *Line scan camera calibration for fabric imaging*. PhD thesis, The University of Texas at Austin, 2012.

[42] C. A. Luna, M. Mazo, J. L. Lázaro, and J. F. Vázquez, "Calibration of Line-Scan Cameras," vol. 59, no. 8, pp. 2185–2190, 2010.