Politecnico di Milano

Dipartimento di Elettronica, Informazione e Bioingegneria

Corso di Laurea Magistrale in Ingegneria Elettronica

2D Position sensing on piston rods for smart hydraulic cylinders

Autore: Alfonso Manuel Campos Alache

Matricola: 804471

Relatore:

Prof. Michele Norgia

Anno Accademico 2015-2016

Ringraziamenti

Desidero esprimere in questa pagina la mia piu' sincera gratitudine a tutti coloro che mi hanno sostenuto in questo lavoro di tesi. In particolare vorrei ringraziare il professore Michele Norgia per i suoi ottimi consigli e per essere sempre stato disponibile durante tutto il mio percorso di tesi. Inoltre vorrei ringraziare il professore Alessandro Pesatori per essere sempre stato disponibile ad aiutarmi quando era necessario.

Un ringraziamento speciale va ai miei carissimi genitori, Mariana ed Alfonso, ed ai miei cari Leslie, Davide e Fausto, che hanno sempre creduto che avrei raggiunto questo importante traguardo e che durante questi lunghi anni non mi hanno mai fatto mancare la loro stima ed il loro affetto.

Infine, vorrei ringraziare a tutti gli amici conosciuti durante il mio percorso di studi. Un percorso difficile ma stimolante che mi ha aiutato a crescere sia dal punto di vista professionale che personale. No, no, no!, non mi sto dimenticando di voi amici miei, che pur non appartenendo alla mia universita', avete supportato le mie assenze, le mie gioie e le mie sofferenze.

Gracias a todos!

Sommario

I sensori mouse ottici sono dispositivi di piccoli dimensioni, economici e capaci di funzionare in assenza di contatto. Inoltre, integrano al loro interno una fotocamera CMOS ed un processore DSP in grado di fornire una misura dello spostamento bidimensionale. Questo lavoro di tesi ha come obbiettivo quello di implementare i sensori mouse in un sistema di misura dello spostamento su pistoni idraulici a doppio effetto, studiando la possibilita' di ottenere un'incertezza nella misura della corsa di circa 1ppm o minore, usando due metodi esperimentali. Il primo metodo compensa la sensitivita' del sensore, che dipende a sua volta dalla velocita' dell oggetto sul quale il sensore acquisisce i fotogrammi ed il secondo metodo compensa direttamente la misura usando riferimenti assoluti posizionati nell'ambiente dove il sensore lavora.

Abstract

Optical mouse sensors small, inexpensive, non-contact devices which integrate a CMOS camera and DSP hardware to provide two dimensional displacement measurements. This thesis work aims to develop a displacement measurement system on double acting hydraulic cylinders, studying the possibility to obtain an uncertainty on the stroke measurement of about 1ppm or slightly below, with two experimental methods. The first method compensates the sensor sensitivity, that depends on the tracking speed where the sensor acquired its frames and the second method directly compensates the measurement, using absolute references placed in the sensor's working environment.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In the last 20 years, the request of innovative vision-based displacement measurement solutions for industrial applications has been rising. This rise depends on the continue evolution in the Computer Vision field. From another point of view, the rise, also depends of the constant needs from manufacturers to be ready and able to fulfill consumer requests, but also to enrich their own known-how, dealing with the day by day market competition. Unfortunately, often these vision systems for displacement purposes cost a lot. And so, researchers see on mouse sensors a feasible solution for measuring displacement in environments where the request of measurement's accuracy is not a priority, obtaining good results.

The present thesis aims to use mouse sensors for measuring two dimensional displacement in industrial applications. Precisely, it focuses in the development of a measuring system for linear and angular position on piston rods with the purpose to obtain smart hydraulic cylinders. This information can be useful for several systems like areal work platforms, hydrostatic steering systems of big machines and others. Finally, it will be proposed a solution for measuring the circumference on big pipes.

The thesis is structured in the following way. In the *First chapter*, will be presented some commercial devices that exists for the position sensing in hydraulic cylinders. Then, in the *Second chapter*, will be introduced the motion estimation theory, that gives the basics to understand how a mouse sensor works. Then, in the *Third chapter*, will be presented the measurement setup. Then, in the *Fourth chapter*, will be explained the calibration method and the experiments done, useful to understand the issues that we had to deal with, during the system development. Then, in the *Fifth chapter*, will be presented the measurement results with his respective algorithms useful for the reduction of the measure uncertainty. Finally,

with the *Sixth chapter*, the thesis will be concluded, giving the general conclusions and some proposes for a further research.

Chapter 2

State of Art

This chapter gives a general explanation of how a hydraulic cylinder works.Then will be shown some devices present in commerce, giving some possible applications of the position sensing in hydraulic cylinders. Finally, the chapter concludes with a briefly review of the necessity to use non destructive testing for specify industrial applications, telling some history about Vision testing.

2.1 Hydraulic cylinders

A Hydraulic cylinder is a mechanical actuator that aims to give a unidirectional force through a unidirectional stroke. It is composed by a cylinder barrel, in which a piston connected to a piston rod moves back and forth. The barrel is closed on one end by the cylinder bottom(cap) and the other end by the cylinder head (gland) where the piston rod comes out of the cylinder. The piston has sliding rings and seals and divides the inside of the cylinder into two chambers, the bottom chamber (cap end) and the piston rod side chamber (rod end or head end).The piston rod also has mounting attachments to connect the cylinder to the object or machine component that it is pushing/pulling.

Generally in a hydraulic system the hydraulic cylinder is the actuator or "motor" side of the system and the "generator" side is the hydraulic pump which brings in a fixed or regulated flow of oil to the hydraulic cylinder, to move the piston. The piston pushes the oil in the other chamber back to the reservoir.

If we assume that the oil enters from cap end, during extension stroke, and the oil pressure in the rod end / head end is approximately zero, the force F on the

piston rod equals the pressure P in the cylinder times the piston area A:

$$F = PxA \tag{2.1}$$

In commerce exists several types of cylinders, but the most common type is:

Double-acting cylinder is a cylinder in which the working fluid acts alternately on both sides of the piston. A double-acting hydraulic cylinder has a port at each end, supplied with hydraulic fluid for both the retraction and extension of the piston and it is used where an external force is not available to retract the piston or where a high force is required in both directions of travel.

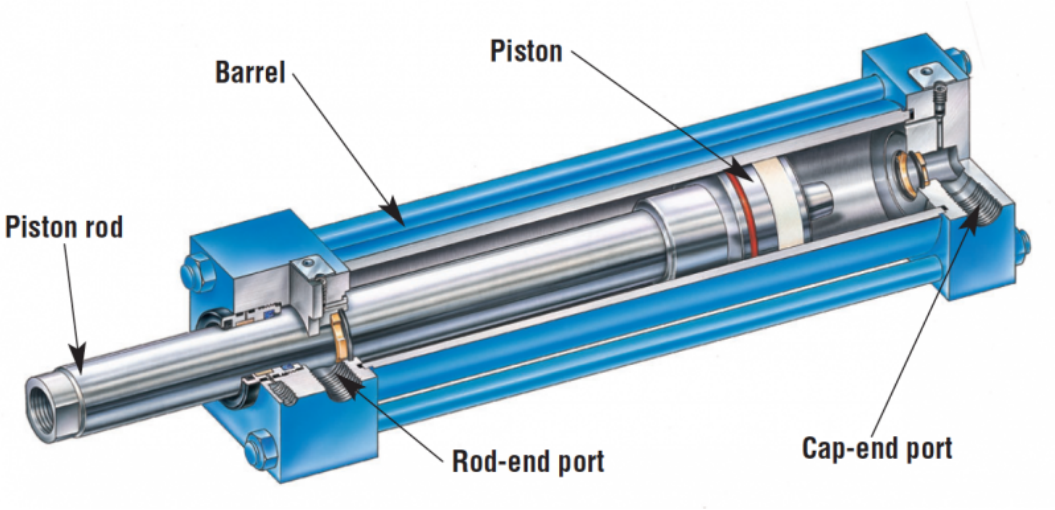


Figure 2.1: Double-acting hydraulic cylinder

In order to connect the piston in a double-acting cylinder to an external mechanism, a hole must be provided in one end of the cylinder for the piston rod and this is fitted with a gland or 'stuffing box' to prevent escape of the working fluid.

2.2 Position-sensing hydraulic cylinder

The position-sensing feature in a hydraulic cylinder provides instantaneous analog or digital electronic position feedback information from the cylinder that indicates the amount of rod extension throughout the range of stroke. This feature can

be used in many applications, notably in construction equipment (engineering vehicles), manufacturing machinery, and civil engineering.

To obtain this position feedback information, in commerce exists two types of approaches: The first one aims to obtain the position modfying the cylinder structure and the second one without this modification.

2.2.1 Internal LDT

In-cylinder linear displacement transducers (LDTs) have been used with limited success in mobile equipment to achieve these goals. A limitation to most in-cylinder LDTs is that the hydraulic cylinder's piston rod must be bored through its center to accommodate certain elements of the LDT, usually the waveguide tube of a magnetostrictive transducer.

The magnestostrive technology is used by the MTS Sensors Corporation, a leading supplier of intelligent hardware and software products in the fields of test and simulation systems and in measuring and automation technology. Among their available products, present in commerce, there is the Temposonics® R series position sensors[13] that exploits the the time-based magnetostrictive position sensing principle.

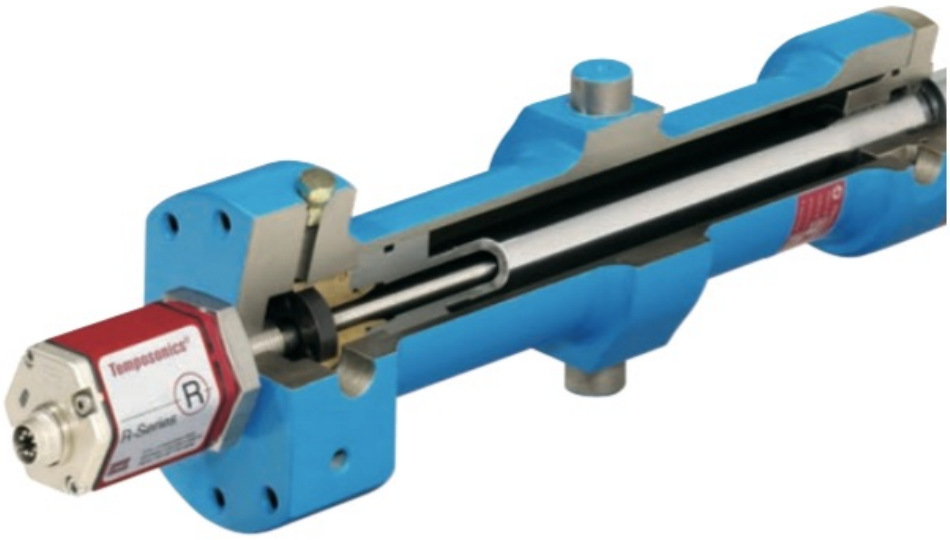


Figure 2.2: Temposonics® R series position sensor

The heart of the time-based magnetostrictive principle is the ferromagnetic mea-

suring element, also known as the waveguide, and a movable position magnet that generates a direct-axis magnetic field in the waveguide. When a current or interrogation pulse passes through the waveguide, a second magnetic field is created radially around the waveguide. The interaction between the magnetic field in the waveguide and the magnetic field produced by the position magnet generates a strain pulse which travels at a constant ultrasonic speed from its point of generation, the measurement point, to the end of the waveguide where it is transformed into an electric pulse in the sensor element. The position of the magnet is determined by measuring the elapsed time between the application of the interrogation pulse and the arrival of the resulting strain pulse with a high-speed counter. The elapsed time measurement is directly proportional to the position of the permanent magnet and is an absolute value. Therefore, the sensor's output signal corresponds to absolute position, instead of incremental, and never requires recalibration after a power loss. Absolute, non-contact sensing eliminates wear, and guarantees the best durability and output repeatability.

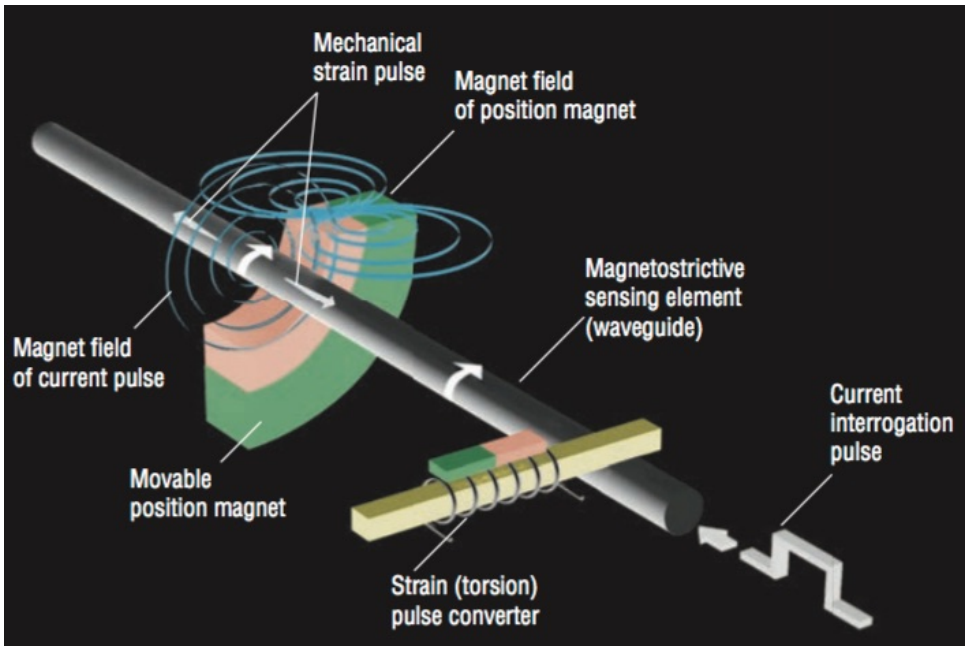


Figure 2.3: Time-based magnetostrictive position sensing principle

The machining and additional production steps associated with "gun drilling" the piston rod add substantial cost to the finished cylinder. And although magnetostrictive LDTs provide extremely high accuracy, this accuracy usually is much greater than is needed for most mobile equipment applications.

Installing linear position sensors into hydraulic cylinders has been a sore subject for

cylinder manufacturers for a long time. Extensive fabrication headaches, including gun drilling, the need to stock all different sensor lengths, and handling issues have contributed to industry frustration.

To solve those problems, the Control Products Inc has been developing a new generation of sensors which eliminates the need for gun drilling. Among their available products, is worth mentioning the SL Series Sensors[6], that exploits the Linear variable differential transformer(LVDT) measurement technology.

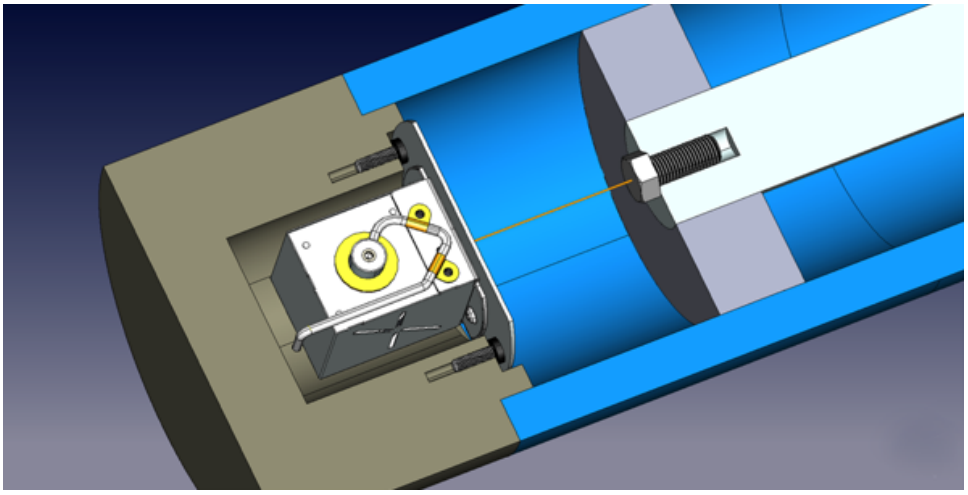


Figure 2.4: SL series position sensor

The heart of SL series is a Linear Variable Differential Transformer(LVDT). It is an absolute position/displacement transducer that converts a position or linear displacement from a mechanical reference (zero, or null position) into a proportional electrical signal containing phase (for direction) and amplitude (for distance) information.

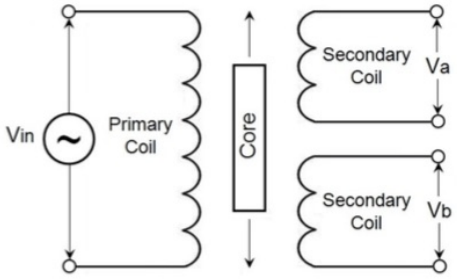


Figure 2.5: Linear Variable Differential Transformer principle

When the primary coil is excited with a sine wave voltage (Vin), this voltage produces a current in the LVDT primary windings, function of the input impedance. In turn, this variable current generates a variable magnetic flux which, channeled by the high-permeability ferromagnetic core, induces the secondary sine wave voltages Va and Vb. While the secondary windings are designed so that the amplitude of the differential output voltage (Va-Vb) is proportional to the core position, the phase of (Va- Vb) with reference to the excitation, called Phase Shift (close to 0 or 180 degrees) determines the direction away from the zero position. The zero, called Null Position, is defined as the core position where the phase shift of the (Va-Vb) differential output is 90 degrees.

The LVDT operation does not require an electrical contact between the moving part (probe or core assembly) and the coil assembly, but instead relies on electromagnetic coupling; this and the fact that LVDTs can operate without any built-in electronic circuitry are the primary reasons why they have been widely used in applications where long life and high reliability under very severe environments.

2.2.2 External LDT

External linear displacement transducers(LDTs) eliminate the need for a hollow hydraulic cylinder rod. Instead, an external sensing bar utilizing Hall-Effect technology senses the position of the piston rod. This is accomplished by the placement of a permanent magnet within the piston.

This Hall-effect technique is used by Rota Engineering Limited to produce external linear position trasducers. Among his available products, they developed the ELA model[18] that eliminates the need for a gun-drilled piston rod.

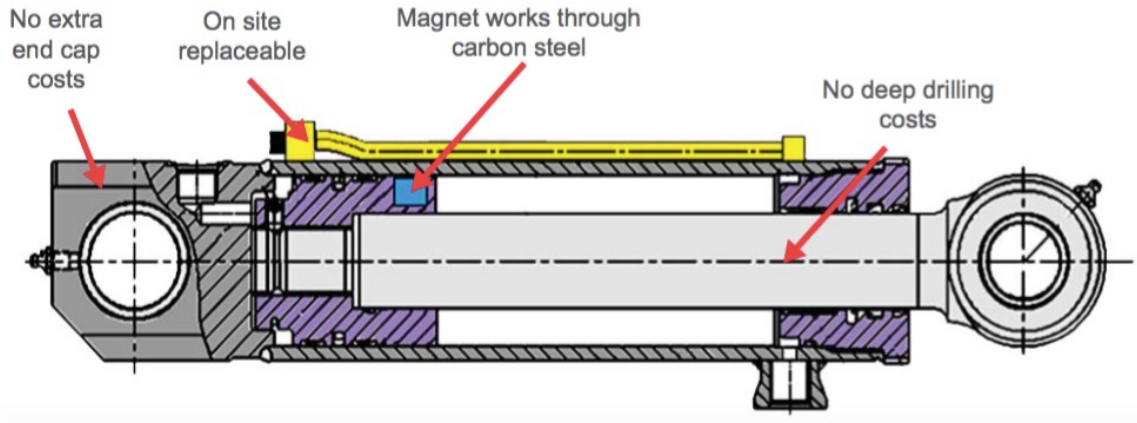


Figure 2.6: ELA position sensor

A transducer is mounted on the cylinder barrel and detects the position of the cylinder's piston by sensing a magnetic field formed by a permanent magnet embedded in the piston. As the piston rod extends or retracts, the magnetic field propagates through the standard carbon steel cylinder wall to communicate with the linear transducer.

The external LDT has the following advantages over the internal LDT:

- Lower overall cost
- Full rod strength is maintained
- The cylinder is easier to assemble, install, and service
- The permanent magnet in the piston should never need to be replaced
- The external sensor is readily accessible and easy to replace
- The sensing bar is small, and its location along the outside of the hydraulic cylinder wall minimizes the potential for environmental damage
- Should machine power be lost and then recovered, the sensor will send the current position
- OEM's can prepare hydraulic cylinders with magnets so end users can add sensing at a later time, if so desired

In addition, certain sensing technologies exhibit greater long-term reliability than others. Internal sensors represent the most reliable solutions. In addition, external cables or wires may be adversely affected by ice, bush and tree limbs, or any other external obstructions that may be encountered.

2.3 Applications

Aerial work platforms is a mechanical device used to provide temporary access for people or equipment to inaccessible areas, usually at height. There are distinct types of mechanized access platforms.The key difference is in the drive mechanism which propels the working platform to the desired location. Most are powered by either hydraulics or possibly pneumatics. The different techniques also reflect in the pricing and availability of each type.

In these case our device can be used to control the load distribution in lifting and downshift stages.

Excavators are heavy construction equipment consisting of a boom, stick, bucket and cab on a rotating platform known as the "house". All movement and functions of a hydraulic excavator are accomplished through the use of hydraulic fluid, with hydraulic cylinders and hydraulic motors. Due to the linear actuation of hydraulic cylinders, their mode of operation is fundamentally different from cable-operated excavators.

In these case our device can be used for safety reasons offering the advantage of operate without the direct viewing of the load, relying to the acquired information arrived in cab, maintaining the load on safety, the machine and the workers on board.

Hydrostatic steering systems Hydrostatic steering stands for any of various steering system configurations where a vehicle is steered solely by means of a hydraulic circuit comprising, as a minimum, a pump, lines, fluid, valve, and cylinder (actuator). that is to say, the vehicle is steered purely by a hydraulically powered steering cylinder. Hydraulic steering has been used forever on a huge number and variety of pieces of equipment - from small forklifts and garden tractors to combine harvesters, large tractors, massive earth moving equipment, construction and mining equipment, aircraft, boats, ships, and many many others.

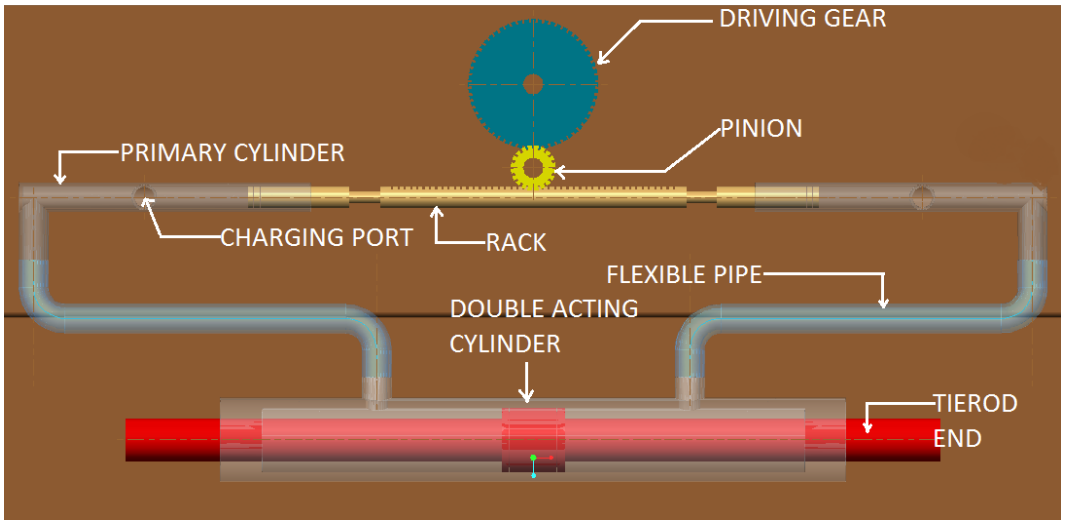


Figure 2.7: Balanced double acting cylinder in a hydrostatic steering system

Most piston-type actuating cylinders used in this steering system are balanced double acting cylinders. In this typology the piston rod extends through the

piston and out through both ends of the cylinder. In a hydro steering system both ends of the piston rod will likely be attached to a mechanism to be operated. The cylinder provides equal areas on each side of the piston. Therefore, the same amount of fluid and force is used to move the piston a certain distance in either direction.

This means that in a hydro steering system, use of the balanced cylinder results in equal steering speed, effort, and number of turns to lock whether turning left or right.This fact, and the units compact design makes it by far the best choice for a hydro steering system.

So in this case our device can be used for to control the steering in machines. It can be done by placing on both sides of the balanced double acting cylinder our device to obtain the instantaneous position as a feedback.

2.4 Non destructive testing

Nondestructive testing (NTD) is the process of inspecting, testing, or evaluation materials, components or assemblies for discontinuities, or differences in characteristics without destroying the serviceability of the part or system.

In contrast to NDT, other tests are destructive in nature and are therefore done on a limited number of samples, rather than on the materials, components or assemblies actually put into service. These destructive tests are often used to determine the physical properties of materials such as impact resistance, ductility, yield and ultimate tensile strength, fracture toughness and fatigue strength, but discontinuities and differences in material characteristics are more effectively found by NDT.

Today modern nondestructive tests are used in manufacturing, fabrication and in-service inspections to ensure product integrity and reliability, to control manufacturing processes, lower production costs and to maintain a uniform quality level. During construction, NDT is used to ensure the quality of materials and joining processes during the fabrication and in-service NDT inspections are used to ensure that the products in use continue to have the integrity necessary to ensure their usefulness.

Most frequently used test methods are:

- Magnetic Particle Testing (MT)
- Liquid Penetrant Testing (PT)

- Radiographic Testing (RT)
- Ultrasonic Testing (UT)
- Electromagnetic Testing (ET)
- Visual Testing (VT)

2.4.1 Visual testing

The history of vision testing probably coincides with the born of Computer vision. Computer vision aims to build autonomous systems which could perform some of the tasks which the human visual system can perform and even surpass it in many cases.

The previous history of Computer vision started with the research of the American pyschologist James Gibson in 1950, who introduced optical flow and based on his theory, mathematical models for optical flow computation on a pixel-by pixel basis are developed. Another fundamental event was the research of the American scientist Larry Roberts in 1960, who in his Ph.D thesis discussed the possibilities of extracting 3D geometrical information from 2D views.

Later, in 1978, a major breakthrough was made by the American neuroscientist David Marr, who created a bottom-up approach scene understanding through computer vision. Low-level image processing algorithms are applied to 2D images to obtain the "primal sketch" (feature extraction of the scene), from which a 2.5 D sketch of the scene is obtained using binocular stereo (provide depth). Finally, high-level (structural analysis, a priori knowledge) techniques are used to get 3D model representations of the objects in the scene. This is probably the most influential work in computer vision ever.[12]

Today many challenges still exist in the development of machine vision systems. The commonly accepted "bottom-up" framework developed by Marr is being challenged, as it has limitations in speed, accuracy, and resolution. Many modern machine vision researchers advocate a more "top-down" and heterogeneous approach, due the difficulties Marr's framework exhibits. A new theory, called "Purposive Vision" is exploring the idea that you do not need complete 3D object models in order to achieve many machine vision goals. Purposive vision calls for algorithms that are goal driven and could be qualitative in nature.

This researches gave the fundamentals to Machine Vision for becoming an important feature of manufacturing. Today machine vision systems provide greater

flexibility to manufacturer's, helping to complete a number of tasks faster and more efficiently than humans alone ever could.

Chapter 3

Motion measurement

The main purpose of this chapter is to introduce a cheap, easy to use, but accurate displacement sensor-system based on visual information acquired from different surfaces. First of all we review the techniques to extract motion from image sequences, then we review the most widely known sensor used for our purposes and finally we review related work in the field of motion measurement, both industrial and academic.

3.1 Motion estimation

Motion estimation is the process of determing motion vectors that describe transformation from one 2d image to another; usually from adjacent images in a video sequence. The motion vectors may relate to the whole image or specific parts and they may be rapresented by a traslational model or many other models than can approximate the motion of a real video camera, such as rotation and traslation in all three dimensions.

Methods for finding motion vectors can be categorised into: [17]

Direct methods Direct or pixel based methods are distinguished by:

- Block-matching algorithm
- Phase correlation and frequency domain methods
- Pixel recursive methods
- Optical flow

Indirect methods Indirect or features based methods use features and match corresponding features between frames, usually with statistical function applied over a local or global area. The purpose of the statistical function is to remove matches that doesn't correspond to the actual motion.

3.1.1 Optical flow

A fundamental problem in the processing of image sequences is the measurement of optical flow or image velocity. The goal is to compute an approximation to the 2D motion field, a projection of the 3D velocities of surface points onto the imaging surface from spatiotemporal patterns of image intensity. Once computed, the measurements of image velocity can be used for a wide variety of tasks ranging from passive scene interpretation to autonomous, active exploration. Of these, tasks such as the inference of egomotion and surface structure require that velocity measurements be accurate and dense, providing a close approximation to the 2D motion field. In other words, Optical flow is the pattern of apparent motion of objects, surfaces, and edges in a visual scene caused by the relative motion observer (an eye or camera) and the scene.

To estimate motion, the idea is to compare consecutive images of a scene produced by a camera and calculate a vector field for each image which shows the displacements of the pixels to get the next image of the scene. This vector field is called optical flow field.

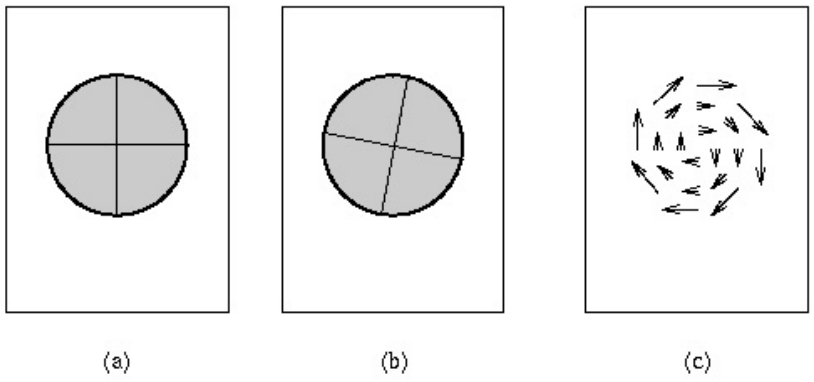


Figure 3.1: The principle of Optical flow: (a) Image at time t, (b) Image at time t+dt, (c) Optical flow field

Differential techniques Differential techniques compute velocity from spatiotemporal derivatives of image intensity or filtered versions of the image.

Since the first algorithm was presented in 1980 by Horn and Schunck[10], several methods have been published to determine optical flow field. The common base of these methods is the *optical flow costraint*, which presumes that the related points in the consecutive images have the same intensity value, with the costraint to be time-invariant and projection-invariant in the image plane:

$$I(x(t+dt), y(t+dt), t+dt) = I(x(t), y(t), t) \tag{3.1}$$

$$\frac{dI}{dt} = 0 \tag{3.2}$$

where I(x,y,t) is the intensity of the (x,y) point in time t.

From Taylor expansion of (2.1) or from the dependencies of the total or parcial derivative using (2.2) the general form of costraints is derived:

$$\frac{dI}{dx}\frac{dx}{dt} + \frac{dI}{dy}\frac{dy}{dt} + \frac{dI}{dt} = 0 \tag{3.3}$$

where $\frac{dx}{dt}$ and $\frac{dy}{dt}$ are the unknowns coordinates of the velocity vector, $\frac{dI}{dt}$ is the time change of the intensity value, $\frac{dI}{dx}$ and $\frac{dI}{dt}$ are the components of the spatial gradient vector of intensity field.

This costraint is not sufficient to determine both components of velocity vector, only the component of local gradient can be estimated. As a consequence to compute the optical flow field is necessary to introduce additional costraints.

The method of Horn and Schunk starts from the observation that the points of the image plane don't move independently, if we view opaque objects of finite size undergoing rigid motion or deformation. Therefore the neighbouring points of moving objects have quite similar velocities and the vectors of the optical flow field vary smoothly almost everywhere. This smoothness constrain represent the following equation:

$$min\left\{\left(\frac{\partial u}{\partial x}\right)^2 + \left(\frac{\partial u}{\partial y}\right)^2 + \left(\frac{\partial v}{\partial x}\right)^2 + \left(\frac{\partial v}{\partial y}\right)^2\right\} \tag{3.4}$$

where u and v are the coordinates of the velocity vector.

Therefore the purpose is to determine a velocity vector field which minimizes the optical flow and the smoothness costraint togheter:

$$min\left\{ \iint_D \left(\frac{\partial I}{\partial x}\frac{\partial x}{\partial t}+\frac{\partial I}{\partial y}\frac{\partial y}{\partial t}+\frac{\partial I}{\partial t}\right)^2+\alpha^2\left(\left(\frac{\partial u}{\partial x}\right)^2+\left(\frac{\partial u}{\partial y}\right)^2+\left(\frac{\partial v}{\partial x}\right)^2+\left(\frac{\partial v}{\partial y}\right)^2\right)dxdy\right\} \tag{3.5}$$

It seems that to compute individually velocity vectors it is necessary to take the whole image into consideration, because every vector depends on the others vector. This method is classified as a ***Global technique***.[2]

Another approach presented by Lucas and Kanade assumes the velocities are the same in a small local area, so defined as a ***Local technique***. [3] Therefore to calculate the velocity vector of a point it is possible to write more than one optical flow constraint because the points in the small region have the same velocity:

$$\boldsymbol{W}\boldsymbol{A}\boldsymbol{v} = \boldsymbol{W}\boldsymbol{b} \tag{3.6}$$

where:

$$\boldsymbol{A} = \begin{bmatrix} \nabla I(x_1, y_1) \\ \nabla I(x_2, y_2) \\ \vdots \\ \nabla I(x_{mxm}, y_{mxm}) \end{bmatrix}; \boldsymbol{v} = \begin{bmatrix} u \\ v \end{bmatrix}; \boldsymbol{b} = \begin{bmatrix} I_t(x_1, y_1) \\ I_t(x_2, y_2) \\ \vdots \\ I_t(x_{mxm}, y_{mxm}) \end{bmatrix}; \boldsymbol{W} = \begin{bmatrix} w_1 & 0 & 0 & 0 \\ 0 & w_2 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & w_{mxm} \end{bmatrix}$$

In this case the local region has m x m points and **W** is a weight matrix.

Because the equation system is over costrained and has no solution therefore the velocity estimates are computed by minimizing

$$\sum_{x\in\Omega(mxm)} \boldsymbol{W}^2(\boldsymbol{x})[\nabla I(\boldsymbol{x})\cdot\boldsymbol{v} + I_t(\boldsymbol{x})]^2 \tag{3.7}$$

After using the least mean squares method, the solution is the following:

$$\boldsymbol{v} = (\boldsymbol{A}^T\boldsymbol{W}^2)^{-1}\boldsymbol{A}^T\boldsymbol{W}^2\boldsymbol{b} \tag{3.8}$$

This method can only measure relatively small displacements therefore it is often called the iterative Lucas-Kanade algorithm.

The previous two algorithms are directly based on the gradients of scenes therefore these techniques are often called *Differential methods*. Unfortunately these techniques suffer from a serious disadvantage: accurate numerical differentiation is sometimes impractical because of small temporal support (only a few frames) or poor signal-to-noise ratio.

Region-based techniques Accurate numerical differentiation may be impractical because of noise, because a small number of frames exist or because of aliasing in the image acquisition process. In these cases dfferential approaches may be inappropriate and it is better to turn to region based matching.

Region-based techniques define velocity vector **v** as the shift **d**=(dx, dy) that yields the best fit between image regions at different times. Finding the best match amounts to maximizing a similarity measure (over **d**), such as the normalized cross correlation or minimizing a distance measure, such as the sum of squared difference (SSD).

The optical flow constraint (namely the related points in consequent images have the same intensity value) can also be found in these techniques indirectly because the best match tries to minimize the difference of the intensity values of the points.

One of the techniques belonging to this group is published by Anandan in 1987 [1], which combines the Laplace-pyramid (to decrease the correlation between the pixels of the images) and the "coarse-to-fine" SSD matching method. Another region-based algorithm presented by Singh [19] is also built on the SSD metric but uses three consequent images from the scene to calculate the displacement of the regions in the second image. Therefore the inaccuracy caused by noises and periodical texture is decreased.

Frequency-domain techniques A third class of optical flow techniques is based on the frequency domain of the image-sequence. One of the advantages brought by these methods is that motion-sensitive mechanisms operating on spatiotemporally oriented energy in Fourier space can estimate motion in image signals for which matching approaches would fail. A good example is the motion of random dot patterns, which are difficult to capture with region-based or differential methods, whereas, in frequency domain, the resulting oriented energy may be rapidly extracted to determine optical flow field.

These methods can be classified in two groups: energy-based approaches are built on the amplitude, phase-based techniques use the phases of the Fourier space to determine the optical flow field. The method developed by Heeger in 1988 [9], formulated as a least square fit of spatiotemporal energy to a plane in frequency space belongs to the first group. An example for the phase-based methods is the algorithm by Fleet and Jepson in 1990.[8]

3.2 Optical mouse

This section focuses only on the most widely common known example of optical flow sensor: *The optical mouse*.

The first mouse was invented in 1964 with the aim to traduce hand movements into cursor movements on the screen. First models had only two mechanical spheres connected to an encoder for measuring the 2D displacement. The major disadvantage of this system was related to the measurement inaccuracy, caused by the uncertancies on the sphere diameter and his contact point with the surface. Furthermore, the system was prone to get dirty easily, slippage and aging. However, these disadvantages were more or less corrected by the user feedback, who corrects the mouse position looking to the cursor movement on the screen. Therefore the mouse trajectory could differ from the cursor trajectory, but this problem produced only a lower ergonomy. To increase the comfort, manufacturers thinked about increasing the sensor sensitivity from the initially 160 cpi (0,158 mm/count) to 1600 cpi (0,0158 mm/count) of successive versions.

With the aim to solve the previous encountered problems, in 1999, the Agilent Technologies produced the first *optical mouse*. The system was composed by a complete image processing system, a high speed but low resolution video camera and an illumination system.

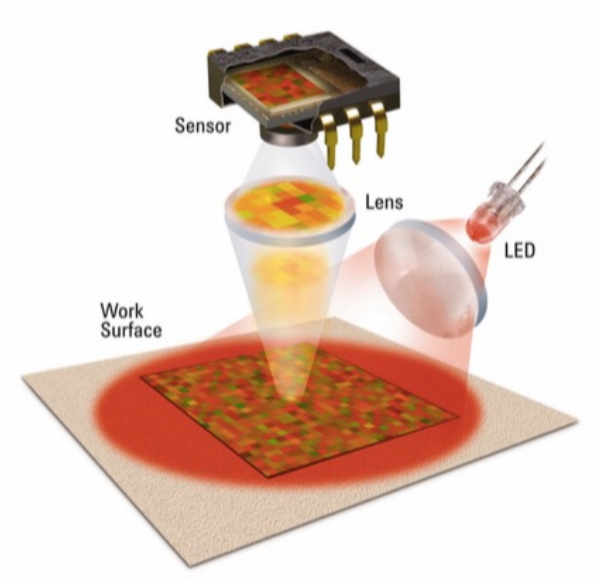


Figure 3.2: The principle of Optical mouse

The working principle of the sensor was to compute the optical flow with a region-based method, known as Digital Image Correlation (DIC). The video camera captures sequentially surface images with a high frame rate and the DSP compute, with correlation based algorithms, the relative displacement between the images.

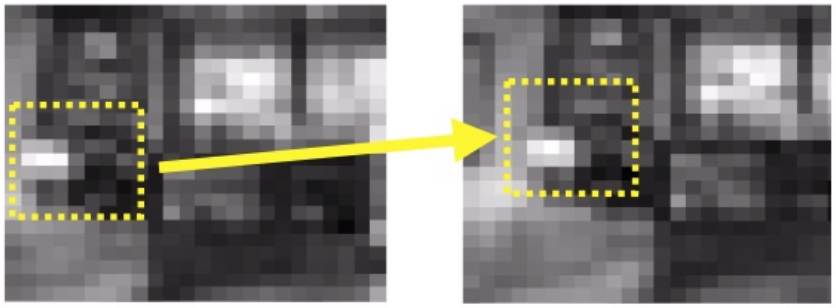


Figure 3.3: Image fingerprints differences under mouse translation.

Through the DIC method, the optical navigation sensor identify common frame regions between images and calculate the distance between them. Therefore, the information is translated to the X and Y coordinates that indicates the mouse movement.

The advantages of this sensor are:

- *No slippage and no aging*, due to surface contactless.
- *Low cost*, due to high density integration offered by CMOS technologies.

Due to these major advantages, mechanical mouse had been sustitued by the optical one.The continue researching introduce, in addition to performance enhancement also a customization for different consumer needs. Modern mouses has achieved excellent performances in terms of measurement speed, resolution, and precision.

The optical mouse uses two distinct but essentially similar techniques for displacement calculation. The classical method uses LED illumination and relies on the micro texture of the surface. The more advanced method is laser speckle pattern technology. Laser speckle patterns can be observed when a rough surface (rough, relative to the wavelength) is illuminated with a coherent light and the interference of the reflected light waves creates a surface dependent random intensity map on the detector. When the detector is moved relative to the surface, the speckle pattern changes accordingly and optical flow can be calculated. The advantage

over surface texture based methods is its accuracy and ability to function properly on relatively textureless smooth surfaces.

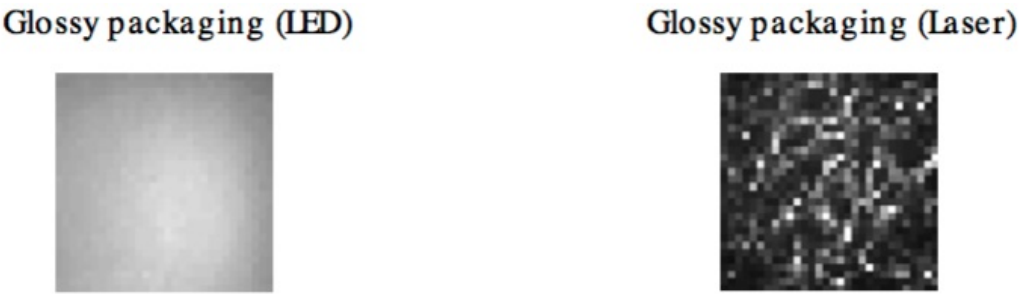


Figure 3.4: Laser uncovers features not detected by Led

Frequency analysis is a less frequently used method. The light reflected from the surface travels through an optical grating, and is focused on a pair of photodetectors. The surface elements, passing in front of the grating generate a certain signal frequency in the detectors depending on the sampling frequency, ground speed, grid graduation, ratio of the image, size of the surface elements and the size of the picture on the grating. The difference of the two signals is computed and the frequency of the difference signals corresponds to the true ground displacement.

However the illumination technique has been chosen between laser or led, it allows to illuminate the tracking surface that reflects the radiation and thanks to a custom designed optical system, the light is focalized in the video camera. The video camera can be CMOS or CCD and it doesn't have color filters, therefore the acquired images are monochromatic. It means that the acquired information become only from the light intensity reflected on the tracking surface. For that reason the illumination system has a fundamental role for the correct system operation.

Generally, laser based sensors are faster than led based sensors in terms of computational speed because a higher radiant flux(power), that allow faster light integration times by the sensor, increase the frame rate.

Finally, the optical system is composed by:

- a *lens*, to focalize the image on the sensor
- a *diaphragm*, to stop the passage of light, except for the light passing through the aperture.
- an *optical subsystem*, to illuminate the surface laterally.

The lateral illumination has a fundamental role on the correct system operation. This solution tends to increase the angle differences of the reflected beam between the various surface points, depending on the surface roughness, maximizing the contrast.

The acquired frames can varies from 16x16 pixel to 36x36 pixel. The image acquisition system has frame rates that arrives up to 12000 fps and in the time between 2 frame acquisitions the DSP compares two images and gives back the relative displacement in terms of counts. To remain at high frame rates, the algorithm selects only a window from the entire image (generally 5x5 pixels), that is then compared with the previous image. Then, the reference window is translated in x and y directions in order to analyze all possible correlations with the previous image. Finally, when the algorithm finds the best matching, the translation is traduced in terms of displacement and send to the computer.

Several optical mouses in commerce use the same working principle but often differs for some characteristics like frame rate, speed, acceleration, resolution, and sensor responsivity.

3.2.1 Digital image correlation method

Image correlation method is based on mathematical calculations in which the variables are the pixel intensity values from the whole acquired images. Let's suppose to calculate the image correlation between two consecutive images defined as: $f(x, y)$ with MxN dimension and $w(x, y)$ with KxL dimension, with $K < M$ and $L < N$. The correlation between $f(x, y)$ and $w(x, y)$ evaluated at a point (i, j) is given by:

$$C(i,j) = \sum_{x=0}^{L-1} \sum_{y=0}^{K-1} w(x+i, y+j) f(x,y) \tag{3.9}$$

where:

$$i = 0, 1, ..., M-1$$

$$j = 0, 1, ..., N-1$$

and the sum is calculated in the region where the first image w and the second image f overlaps.

The maximum value of $C(i, j)$ indicates the best matching between the first image $w(x, y)$ and the second image $f(x, y)$. If $w(x, y)$ is, for example, a region belonging to the image $f(x, y)$, the maximum value of C will be obtained where the points

(i, j) allows the complete overlap of the region under exam with his correspond copy in the image $f(x, y)$.

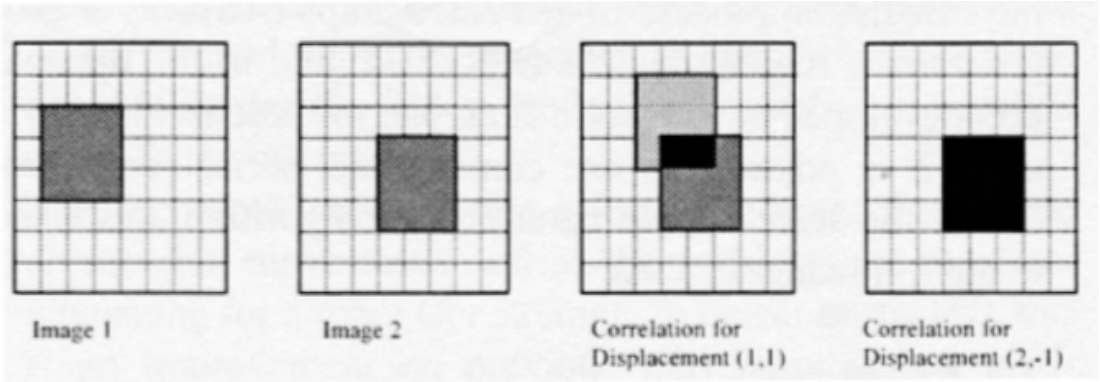


Figure 3.5: Image correlation method

This method is very sensible to pixel intensity variations, because if the intensity of $f(x, y)$ doubled, the same will occur with the correlation $C(i, j)$. To solve this problem, is better to use the Pearson or correlation coefficient (r), that normalize the $C(i, j)$ value and make it independent to pixel intensity variations from the image $f(x, y)$ and the window $w(x, y)$.

For monochromatic images, the Pearson coefficient(r) is defined as:

$$r = \frac{\sum_i (x_i, x_m)}{\sqrt{\sum_i (x_i, x_m)^2} \sqrt{\sum_i (y_i, y_m)^2}} \tag{3.10}$$

where x_i and y_i are respectively, the pixel intensity from the first and the second image, and x_m and y_m are respectively, the average pixel intensity from the first and the second image.

For example if r=1 both images are identical, if r=0 both images are totally different and if r=-1 both images are anticorrelate (one is the negative of the other one). Usually the correlation coefficient is used to compare two images belonging to the same object, acquired at different times, to determinate an eventually displacement. Generally is impossible to make a good noise filtering while the image is acquired, so a good solution is to define a minimum r value beyond the image can't varies. Depending on the application this threshold can be fixed in a range that varies from 0.85 and 0.30.

The advantage of correlation method is that a simple calculation of C(i,j) allows to determinate the displacement (i,j) of an object in the whole image. Everything

is done without contact with the tracking surface. Unfortunately this method has some disadvantages that have to be solved or mitigated:

First, the need of a hardware with enough memory and computational capacity to store two images and to do all necessary calculations each time.

Second, with the previous formulas (2.9) and (2.10), the computational time becomes proportional to the square root of the area, where the correlation is done.

Third, there is a high sensibility to:

- *alignment errors*
- *distorsion*
- *vignetting*, a reduction of an image's brightness or saturation at the periphery compare to the image center

Fourth, the need to have a high contrast. If the image intensities were uniform is not possible to obtain a coherent result.

Fifth, the intensity variation of the light source between the actual and the successive frame introduce a further noising source.

Finally, another source of error are images with periodic or recursive structures.

3.2.2 Performances

To be oriented with the various available sensors we have to evaluate its technical specifications for selecting the best one for our purpuoses:

1. **Frame rate**

   This parameter gives the number of the acquired and elaborated frames in a second. A higher frame rate, gives a higher sensor speed. In fact, at a given speed, with a very low working frequency the two acquired successive images couldn't have common regions, with a consequent error in measurement. Furthermore, a high frame rate enhance the measurement accuracy because the light integration time becomes lower. Therefore, at a given speed, an acquired image with high frame rate is sharper than an acquired image with low frame rate because the last one tends to be more blurred due to the scene movement during the shutter speed of the image.

2. **Counts per inch(cpi)**

Gives the maximum number of counts that the sensor is able to provide for one inch displacement. In other words, is sensor's resolution in terms of displacement. Actually, what really matters is sensor's pixel number. In fact, at equal number of pixels, the maximum resolution depends of the ratio between the Field of view (FoV) and the sensor area. A wide surface determines a low resolution, but a high image luminosity with a less powerful light source. In the present thesis this parameter is important our pipe application, because I change completely the optical system for measuring displacement at a fixed height far from the pipe surface. Actually, in the rod application i realized that varying this parameter doesn't influence our measurement system.

3. **Inch per second(ips)**

   Gives the maximum sensor speed over a tracking surface, in order to guarantee a good accuracy. Generally it depends on the tracking surface type, and is higher for laser based sensors and for high frame rates. To understand the utility of this parameter is better to explain how the sensor works.

   First of all, the maximum speed depends on the image dimension detected by the sensor, so his Field of view(FoV). Furthermore for the correct working of the correlation algorithm, is important to acquire images with high contrast obtained only with a high light integration time. On the other hand, a high integration time slows down the frame rate and tends to saturate pixels due to high levels of illumination. Therefore the system performance has a strong dependance from the tracking surface quality. Uniform surfaces gives a low contrast, on the other hand low reflectance surfaces need a higher light integration time. These characteristics will be analyzed in detail in next chapters.

   In average, sensors work up to tens of m/s. In the present thesis this parameter is important only for the start/stop phases, but not for the normal operation of the system because hydraulic systems usually work at low speeds. And for pipes application we don't have speed constraints, however a fast measurement is our ideal objective.

4. **Acceleration**

   Gives the maximum sensor acceleration, in order to guarantee a good accuracy. It is given in g terms, therefore the gravity acceleration. In average, led based sensors work up to 20g, and laser based sensors work up to 30g. In the present thesis this is important for the start and stop phases of the motion on our piston application, because we have to control initial accelerations

that depends on external actuators and also on resistance from sealings.

5. **Working temperature**

   This parameter is very important for our applications, because at normal working conditions hydraulic pistons can reach high temperatures. So it should be used sensor's that allow high working temperatures or in alternative techniques to decrease the average temperature. An example is Peltier's cell.

   In average, sensor's working temperatures can varies from 0 to 80 degrees.

3.3 Related work

Optical measurement is an emerging discipline, with existing commercial solutions and research activity, however many problems remain unsolved. As for commercial technologies the most widely known example is the optical mouse, which on the other hand has generated a fair amount of academic research.

Displacement measurement solutions using optical mice were suggested by several authors [16, 4, 20]. T.W. Ng investigated the usability and accuracy of optical mice for scientific measurements in several articles [14, 15] with good results. It was found that the readings possessed low levels of error and high degrees of linearity. The mean square error for measurements in the x-axis increased significantly when the distance between the surface and the detector was increased possibly caused by the illumination direction of the mouse.

Moreover, several authors proposed the use of optical mice as a dead reckoning sensor for small indoor mobile robots in one and two sensor configurations. By using one sensor and kinematical constraints from the model of the platform, a slip free dead reckoning system can be realized. The kinematic constraint originates from the sensors inability to calculate rotation. By using two sensors the constraint can be removed and the measurements become independent of the platforms kinematics. Systematic errors originate from measurement errors, alignment errors and change of distance from the ground. Bonarini, Matteuci and Restelli [4] achieved results comparable to other dead reckoning systems up to a speed of 0.3 m/s. Sorensen [20] found that the error of the two mice system was smallest when the sensors were as far as possible from the centre of rotation, and when good care were taken of maintaining constant height. He found that when these constraints were met, the system performed significantly better than other dead reckoning systems.

In their work Palacin et al, [16] found that if measurements from an array of sensors were averaged the error became independent from the distance traveled. They also found that the sensor needed a different calibration when moving in an arc, possibly due to the sideways illumination used in computer mice. Another problem was the extreme height dependence of the sensor, which made it impossible for them to use it on carpet. They proposed a modified sensor for they found mice to be unfit for mobile robot navigation. The results of the authors of this article were similar to other researchers, they found that one way to make mouse sensors useful for navigation is to equip them with telecentric lens, to avoid magnification changes, to use homogeneous illumination, to avoid directional problems and to use two sensors to get rid of kinematic constraints. Takacs and Kalman [21] by using different magnification larger portions of the ground will be projected on the sensor making higher speeds possible, but this is limited by ground texture.

Mouse sensors are cheap and readily available and with certain modifications they can be used for accurate displacement measurement or low speed mobile robot dead reckoning. However they are limited by their low resolution and speed and their algorithm can only be changed by the factory.

Horn et al. aimed at developing a sensor system for automobiles. They used a fusion approach with two cameras and a Kalman filter. One of the cameras is a forward looking stereo camera to estimate yaw rate and forward velocity, the other camera is facing the ground and used to estimate two dimensional velocity. It was found that the camera facing the ground gave better results for lateral and longitudinal velocity than the stereo camera. The fusion approach provided good results even when one of the sensors was failing. The system was tested at slow (lower than 1 m/s) speeds on a towed cart in a lab. [11] Chhaniyara et al. followed a somewhat similar approach and used a matrix camera facing the ground to estimate speed over ground. They used a mechanism that moved the camera over sand and compared optical flow speed estimates with measurements from an encoder attached to the mechanism. They used Matlab and the Lukas and Kanade algorithm to compute optical flow. They obtained good results at low speeds (0-50 mm/s), however the suitability of the algorithm they used is questionable [5].

This technology has already found its way to the transportation industry as well. Corrsys - Datron has a one-of-a-kind optical speed sensor [7] used for testing the slip-free measurement of transversal dynamics in vehicles before mass production. The new CORREVIT S-350 represents yet another a major step forward in the advancement of optical measurement technology. Due to its considerably extended working range, the S-350 Sensor is ideally suited for application with trucks, busses and off-road vehicles. The sensor is claimed to be working on any surface, including water and snow, but it is priced for the big automotive manufacturers. It uses the

frequency analysis method.

OSMES by Siemens is an optical speed measurement system for automated trains [22]. It uses the principle of laser speckle interferometry mentioned above, and looks directly on the rails to measure the trains speed.

It is clear that much work has been done in the field of optical navigation however several issues remain open for research. Current industrial solutions are somewhat bulky and definitely not priced for the average mobile robot. Solutions by academic researchers have not matured to the level of really useful applications. Mouse chips are the mostly the sensors of choice. With some modifications their problems of ground distance, lighting and calibration can be helped, but their current speed and resolution is simply not enough for high speed (the order of ten m/s) applications.

In conclusion, more work in the area of texture analysis, optics design and image processing hardware is needed.

Chapter 4

Measurement setup

First of all the experimentation initiates with the development of the hardware for the communication between the optical sensors and microcontroller. Then, the development of microcontroller's firmware and software for data communication from microcontroller to pc. Finally, the chapter concludes with the system calibration.

4.1 Optical Sensors

The optical sensors under test are:

1. ADBS 350, an optical finger navigation sensor (OFN)
2. ADNS 9800, a Laserstream gaming sensor (LGS)
3. PMT 9101DM-T2QU, an optical tracking sensor (OTS)

The ADBS A350 sensor is a small form factor LED illuminated optical finger navigation system. It is a low power optical finger navigation sensor with low power architecture and automatic power management modes, making it ideal for battery and power sensitive applications such as mobile phones. Is also capable of high-speed motion detection, up to 20 ips. In addition, it has an on chip oscillator and integrated LED to minimize external components. There are no moving parts, thus provide high reliability and less maintenance for the end user. In addition, precision optical alignment is not required, facilitating high volume assembly.

The ADNS 9800 LaserStream gaming sensor comprises of sensor and VCSEL in a single chip on board package. It provides enhanced features like programmable

frame rate, programmable resolution, configurable sleep and wake up time to suit various PC gamers preferences. The advanced class of VCSEL was engineered by Avago Technologies to provide a laser diode with a single longitudinal and a single transverse mode. Used with his customized optical lens, it provides a complete and compact navigation system without moving part and laser calibration process is not required in the complete mouse form, thus facilitating high volume assembly.

The PMT9101DM-T2QU is an Optical Track Sensor (OTS) using optical navigation technology for motion reporting purposes. The OTS integrates LED source and optical sensor in a single small form factor package. It is built in image recognition engine, which does not require additional hardware or any special markings on surface. Furthermore, offers a direct SPI output together with motion interrupt signal, provides easy integration with host system.

In the following table (4.1), their most important features are summarized:

Sensor's features			
Sensor	OFN	LGS	OTS
Max speed(m/s)	0,5	3,81	3,81
Max acceleration(g)	x	30	30
Resolution(cpmm)	4,92	7,87	3,94
Temperature(C)	-20 to 70	0 to 40	0 to 40
Pixel matrix(pixel)	19 x 19	30 x 30	36 x 36
Illumination source	Led IR	Laser IR	Led IR

Table 4.1: Comparison of principal sensor's features

With a experimental method, that will be explain in the next chapter, the best choice for our purpuoses is the OTS sensor.

4.2 Microcontroller

The principal role of the microcontroller is to communicate with the sensors under test and to read the necessary data that next is going to be send to the computer for a successive processing.

Due to our needs of doing basics operations, was chose the mbed NXP LPC1768 prototyping board. It allows to make a rapid prototyping without having to work with low level microcontroller details. Furthermore, its possible to compose and compile embedded software using a browser-based IDE, then download it quickly and easily, using a simple drag-and-drop function, to the board's NXP Cortex-M3

microcontroller LPC1768. Therefore it can be used to be more productive in early stages of development.

The microprocessor is an ARM Cortex-M3 device that operate at up to 100 MHz. The available serial peripherals are SPI, I^2C, Ethernet, USB, CAN and a UART module that allow the communication with pc. There are 512 KB of Flash memory and 64 KB of SRAM. The architecture uses a multi-layer bus that allows high-bandwidth peripherals such as Ethernet and USB to run simultaneously, without impacting performance. Furthermore, it has analog peripherals as 12 bit ADC with 8 channels and 10 bit DAC. Other peripherals like general purpose DMA, motor control PWM, quadrature encoder interface to support three-phase motors and 32-bit timers/counters complete this powerful microcontroller. The following figure(4.1) summarizes the LPC1768 architecture:

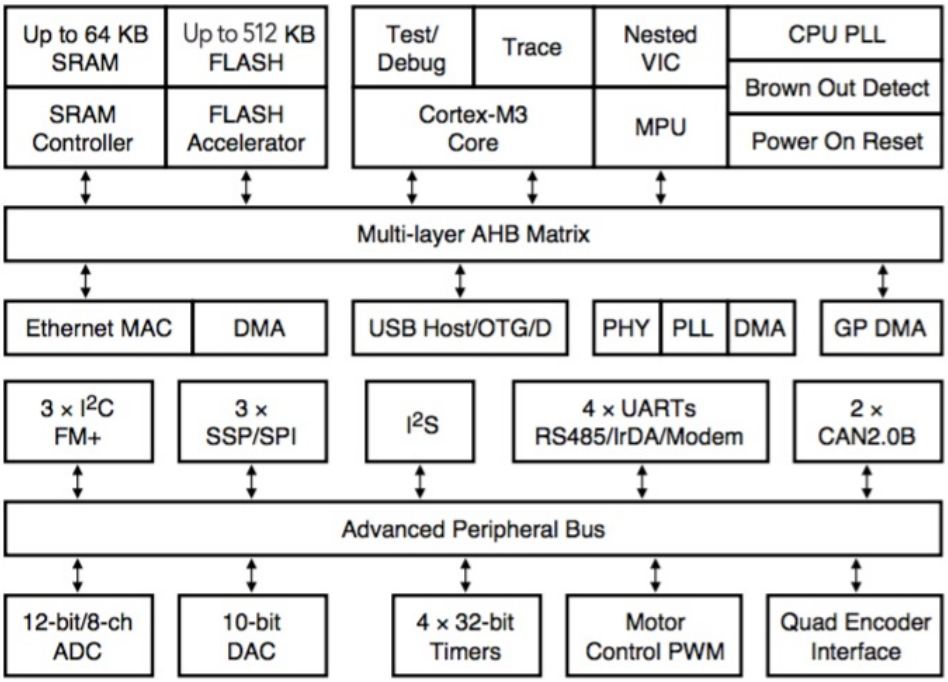


Figure 4.1: LPC1768 block diagram

4.3 Data communication

The data communication from the sensor to the computer use a serial architecture. The first stage, from sensor to microntroller, is managed by an SPI bus and the

second stage, between the microcontroller and computer, is managed by an USB bus with UART module.

4.3.1 Communication with microcontroller

The Serial Peripheral Interface (SPI) bus is a synchronous serial communication interface specification used for short distance, and low speed (1 to 50 Mbit/s) communication. SPI devices communicate in full duplex mode using a master-slave architecture with a single master. The master device originates the frame for reading and writing. Multiple slave devices are supported through selection with individual slave select (SS) lines.

The SPI bus specifies four logic signals:

- SCLK : Serial Clock (output from master).
- MOSI : Master Output, Slave Input (output from master).
- MISO : Master Input, Slave Output (output from slave).
- SS : Slave Select (active low, output from master).

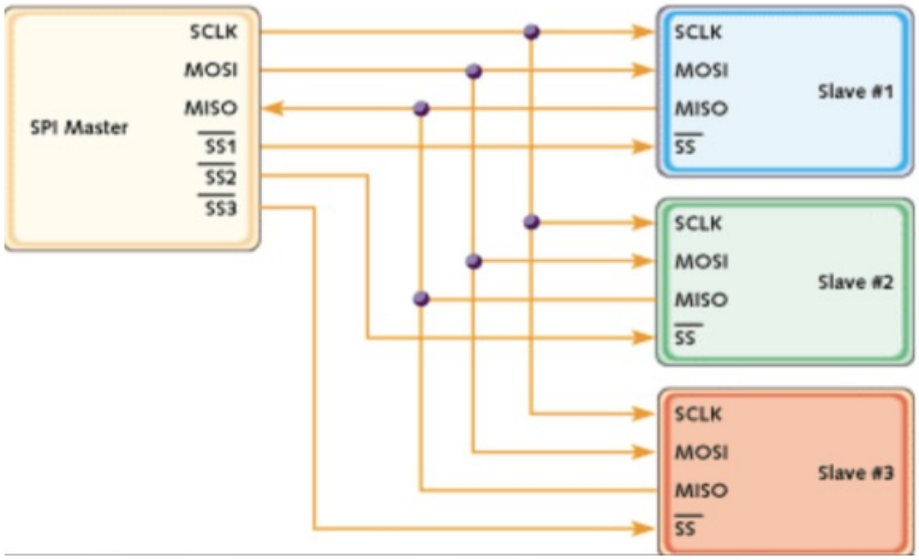


Figure 4.2: SPI bus with three independent slaves

In our case the master is the microcontroller and the slave is the optical sensor. Generally, SPI bus is used to connect microcontrollers/processors to peripherals.

4.3.2 Communication with pc

The microcontroller can communicate with the host PC through an USB Virtual Serial Port over the same USB cable that is used for programming. This allow to communicate with LabView, that is the software for data processing. Then, LabView access to the serial port through its Virtual Instrument Software Architecture(VISA).

The serial communication is full duplex. The transmission is asynchronous and use a hardware module UART (Universal Asynchronous Receiver/Transmitter). UART aims to transmits data organized by words, then transform words in serial bits using for example shift registers, and finally send data through the transmission channel, in this case USB bus. The receiver, thanks to a second UART, converts again the serial bits in words and send those words to the host system.

To control the data flow, each frame has a start bit and a stop bit. To detect eventually errors in data flow, is possible to use parity bits that are send in each frame. All UART's operations are controlled by a clock signal, that works at multiply of transmission rate. As the system is asynchronous, is necessary that the serial communication parameters of transmitter have to be equal to the receiver ones. The parameters are:

- *Speed:* The serial port use two level signals, so the data rate in bit is equal to the symbol rate expressed in baud. The speed goes from 1200 to 115200 bit/s. Due to control bits, only 80 percent of transmitted bits is useful data.
- *Data bit:* The number of bits in the whole word can vary from 5 to 9.
- *Parity bit:* it can be set as none(N), odd(O), even(E), mark(M) or space(S).
- *Stop bit:* It indicates the number of stop bits used in the whole frame.

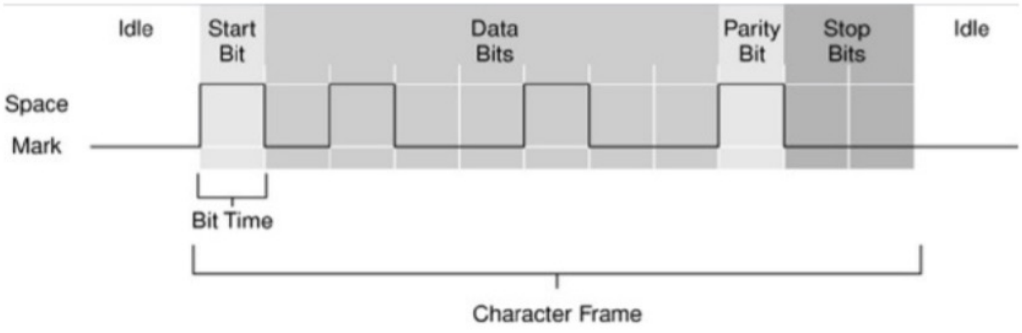


Figure 4.3: Asynchronous serial transmission.

4.4 Hardware

It was design the hardware for the three sensors in order to allow the communication between sensor and the mbed microcontroller. It will be shown the circuit layout and pcb for the OTS sensor. For the other sensors the design was similar.

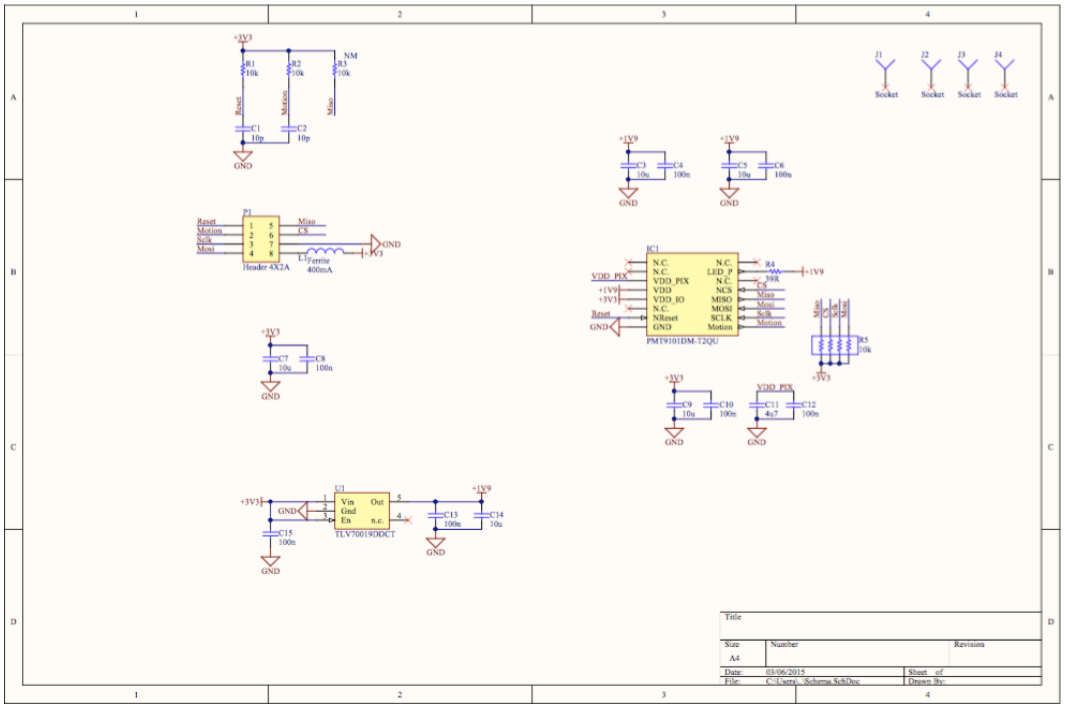


Figure 4.4: CAD for OTS sensor

The out pins are: ncs, miso, mosi, sclk, motion, reset, vdd, gnd and those pins have to connect, through a flat cable, to the mbed microcontroller input pins 3,4,5,6,7,8. With the 3,4,5 pins reserved for mosi, miso, sckl, the pin 7 for ncs, pin 8 for reset and the vdd and gnd are connected to its correspond pins present in mbed microcontroller. The circuit also has a voltage regulator with a voltage of 2 volts, necessary to power up the system.

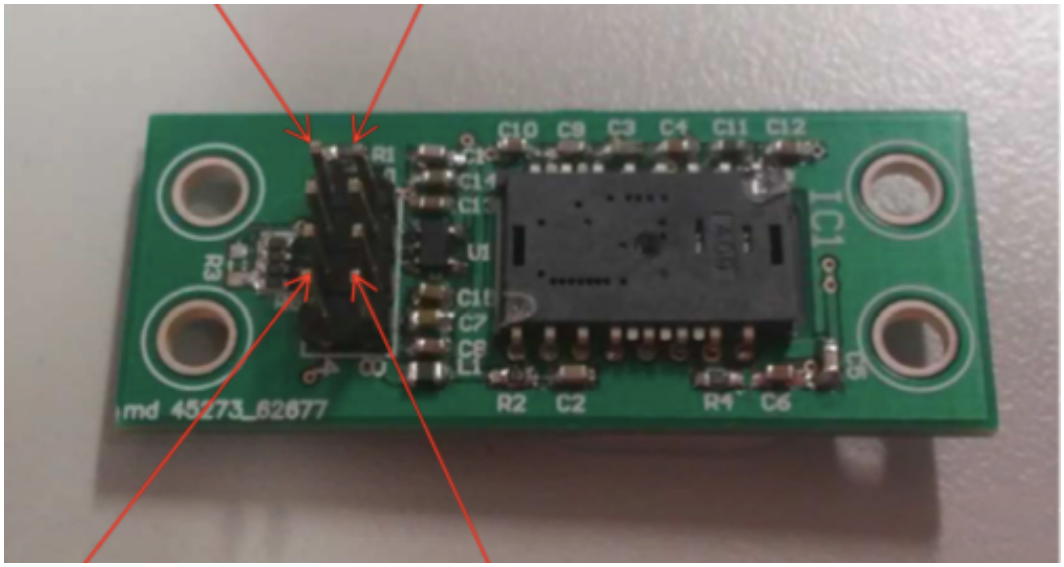


Figure 4.5: PCB for OTS sensor

4.5 Firmware and Software

To extract the useful data from the sensor were developed two programs. The first one, for displacement acquisition and the second one, for frames acquisition.

4.5.1 Displacement program

The displacement program is divided in two stages:

The first stage, is the microcontroller firmware that communicates with the sensor through an SPI bus. The aim of this program is to read the relative displacement of the X and Y axis. This information in given in count dimension. Moreover, there is the need to read the Shutter speed or exposure time. It is defined as the length of time when the sensor inside the camera is exposed to light. This amount of light that reaches the image sensor is proportional to the exposure time. This parameter is going to be used for aplying the algorithms that will be presented in the next chapter.

So the microcontroller reads those registers, with an specified sampling time, in the following way:

- *Motion,* provide a signal when an eventually motion is detected.
- *Delta_X,* provide the relative displacement in X axis in 2 bit's complement.

- *Delta_Y,*provide the relative displacement in Y axis in 2 bit's complement.
- *Shutter,* is automatically adjusted to keep the average pixel values within normal operating ranges.

The second stage, is the LabView program that receives all the data trough the serial port. So the serial data is read in a continuos way thanks to the API VISA. The program aims to converts the displacement from count dimensions into millimeters. It can be done by dividing the displacement in count dimension by his effective sensitivity(cpmm) parameter. Then, it has to integrate the displacement received each cycle. Finally, it has to apply appropriated algorithms that corrects or compensate the displacement with a defined criteria.

Now, the programs will be presented in the following way: First the microcontroller's firmware and finally the LabView software.

```
#include "mbed.h"
#include "stdint.h"
#define delay 1 //Delay of cycle in ms.

//PORT settings
SPI spi(p5, p6, p7);
DigitalOut ncs(p8);
DigitalIn mot(p9);
DigitalOut nreset(p10);
Serial pc (USBTX, USBRX);

//FUNCTIONS
void write (int address, int data);//write to sensor.
uint8_t read (int address);//read from sensor.

int main(){
uint8_t motion, aux, maschera;
unit8_t shutter_up, shutter_low;
uint8_t deltax_L, deltax_H, deltay_L, deltay_H;

//SERIAL settings
pc.baud(115200);

//SPI settings
spi.format (8,3); // high steady state clock, 2nd edge capture.
spi.frequency (2000000); // spi clock at 2 Mhz
```

```
//Power up and reset sequence
nreset=1;
ncs=1;
ncs=0;
write(0x3A,0x5A);//Power_up register
wait_ms(50);
aux=read(0x02);
aux=read(0x03);
aux=read(0x04);
aux=read(0x05);
aux=read(0x06);

//Registers settings
write(0x0F,0x00);//Config_1 register, set resolution to 100cpi.

//READ motion, delta_X, delta_Y and Shutter registers.
while(1){
    write(0x02,0x00);//clear motion register
    motion=read(0x02);
    mask=(motion)&(128);//mask to control the motion MSB.
    if (mask==128)// if a motion event is detected, so read dx, dy.
    {
        deltax_L=read(0x03);
        deltax_H=read(0x04);
        deltay_L=read(0x05);
        deltay_H=read(0x06);
    }
    else // otherwise put dx, dy to zero.
    {
        deltax_L=0;
        deltax_H=0;
        deltay_L=0;
        deltay_H=0;
    }
    shutter_up=read(0x0C);
    shutter_low=read(0x0B);

    //SEND data to host pc.
    pc.printf("%d",motion);
    pc.printf("\t");//char to separate bytes.
```

```
    pc.printf("%d",deltax_L);
    pc.printf("\t");
    pc.printf("%d",deltax_H);
    pc.printf("\t");
    pc.printf("%d",deltay_L);
    pc.printf("\t");
    pc.printf("%d",deltay_H);
    pc.printf("\t");
    pc.printf("%d",shutter_up);
    pc.printf("\t");
    pc.printf("%d",shutter_low);
    pc.printf("\n");// termination char

    wait_ms(delay); //DELAY of cycle.
}//end while

}//end main

void write(int address, int data){

ncs=0;
spi.write(address+128);//put msb=1 on register address.
spi.write(data);//send data.
wait_us(35);
ncs=1;
wait_us(180);
}

uint8_t read(int address){

uint8_t out;
ncs=0;
spi.write(address);//put msb=0 on register address.
wait_us(160);
out=spi.write(0x00);//read data.
wait_us(1);
ncs=1;
wait_us(20);
return out;
}
```

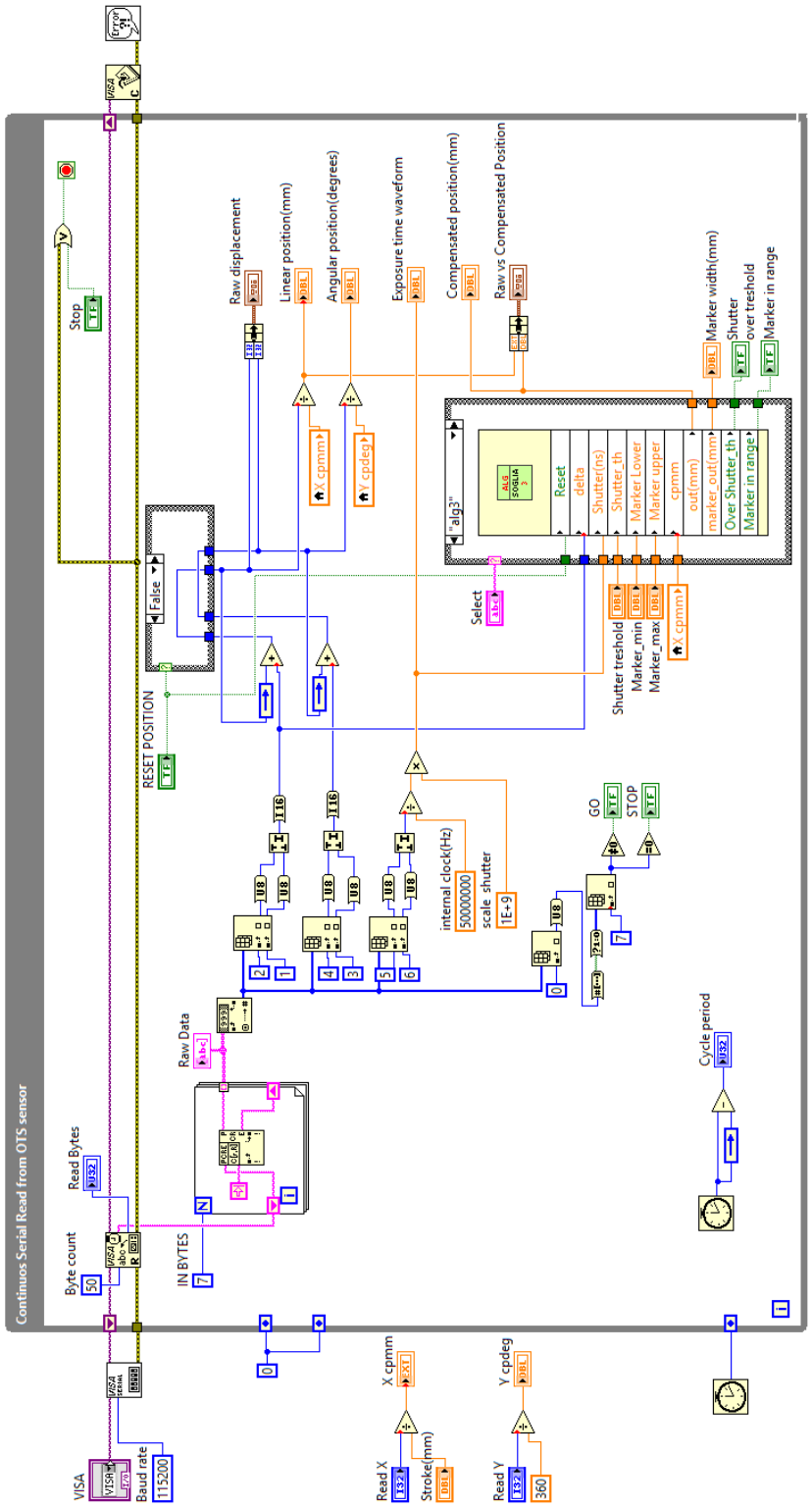


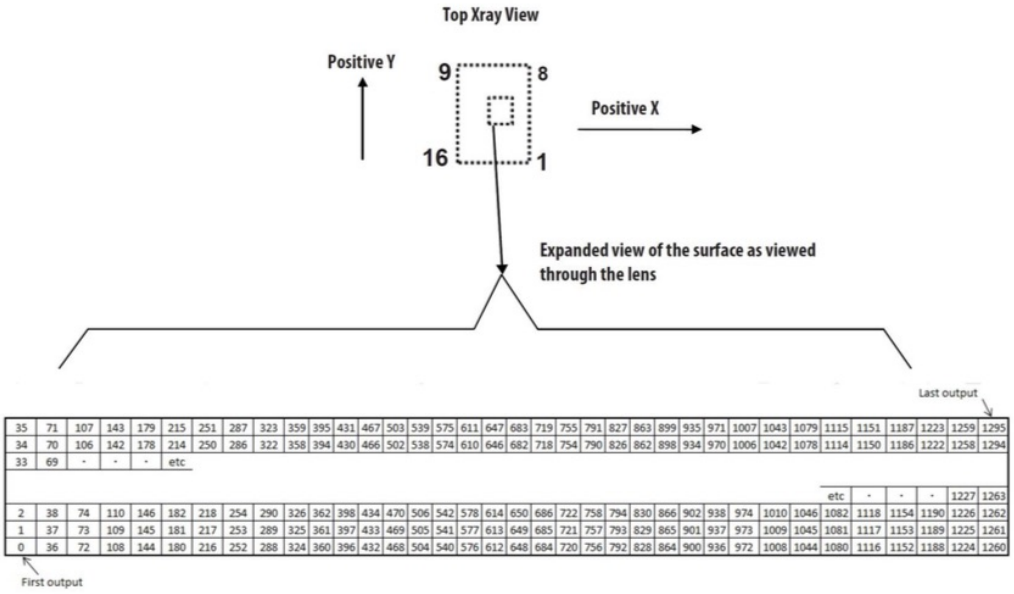
Figure 4.6: Main Labview program: Block diagram

4.5.2 Frame acquisition program

The frame acquisition is useful for calibration and also to understand the limits of maximum speed and resolution of the system. It becomes useful for the experiment where we change the optics.

Also the frame acquisition program is divided in two stages:

The first stage, is the microcontroller firmware. The program aims to read the whole 36x36 pixel matrix, so 1296 pixels/frame. To read the whole frame is used the Frame Capture burst mode.This modality sends the full array of pixel values from a single frame without having to write the register address, where data is stored, to obtain each pixel data. The following figure(4.4) shows the Pixel map in a surface reference.



35	71	107	143	179	215	251	287	323	359	395	431	467	503	539	575	611	647	683	719	755	791	827	863	899	935	971	1007	1043	1079	1115	1151	1187	1223	1259	1295
34	70	106	142	178	214	250	286	322	358	394	430	466	502	538	574	610	646	682	718	754	790	826	862	898	934	970	1006	1042	1078	1114	1150	1186	1222	1258	1294
33	69	·	·	·	etc																														
																														etc	·	·	·	1227	1263
2	38	74	110	146	182	218	254	290	326	362	398	434	470	506	542	578	614	650	686	722	758	794	830	866	902	938	974	1010	1046	1082	1118	1154	1190	1226	1262
1	37	73	109	145	181	217	253	289	325	361	397	433	469	505	541	577	613	649	685	721	757	793	829	865	901	937	973	1009	1045	1081	1117	1153	1189	1225	1261
0	36	72	108	144	180	216	252	288	324	360	396	432	468	504	540	576	612	648	684	720	756	792	828	864	900	936	972	1008	1044	1080	1116	1152	1188	1224	1260

Figure 4.7: Pixel map in a surface reference

Moreover, other registers, useful for calibration, are going to be read as shutter, squal, average pixel value, maximum pixel value and minimum pixel value.

First of all the transmission initiates with writing on the *Frame_Capture* register to capture the next available complete frame of pixel values that are going to be stored in RAM. Furthermore, this action disable the navigation and a reset sequence is required to restore it. Then, is necessary to read the *Pixel_burst* register until

all 1296 pixels are transferred. Finally, the rest of the registers are read in the following order:

- *Shutter,* as previously explained, it stands for shutter speed or exposure time.
- *Squal,* the surface quality is a measure of the number of valid features visible by the sensor in the current frame.
- *Pixel_sum,* provide the average pixel value in the current frame.
- *Max_pixel,* provide the maximum pixel value in the current frame.
- *Min_pixel,* provide the minimum pixel value in the current frame.

The second stage, is the LabView program. The program aims to read in a continuos way an array of 1301 bytes, where the first 1296 bytes correspond to the pixel values of the whole frame and the last 5 bytes correspond to the calibration data. To reconstruct the frame, the subarray is organized in a 36 x 36 matrix and then transform in an image. In this way, the image and the calibration data are shown in the LabView front panel.

Now, the programs will be presented in the following way: First the microcontroller's firmware and finally the LabView software.

```
#include "mbed.h"
#include "stdint.h"
#define delay 1 // Delay in us
#define n_pixel 1296 // 36x36 pixel matrix

//PORT settings
SPI spi(p5, p6, p7);
Serial pc (USBTX, USBRX);
DigitalOut ncs(p8);
DigitalIn mot(p9);
DigitalOut nreset(p10);

//FUNCTIONS
void write (int address, int data);//write to sensor.
uint8_t read (int address);//read from sensor.

int main(){
//Variables
int i, aux, mask, motion;
uint8_t frame[n_pixel];
uint8_t shutter_up, shutter_low;
```

```
uint8_t squal;
uint8_t av_pixel, max_pixel, min_pixel;

//SERIAL settings
pc.baud(115200);

//SPI settings
spi.format (8,3); // high steady state clock, 2nd edge capture.
spi.frequency (2000000); // spi clock at 2 Mhz

while(1){
    //Power and reset sequence.
    nreset=1;
    ncs=1;
    ncs=0;
    write(0x3A,0x5A);//Power up reset register
    wait_ms(50);
    aux=read(0x02);
    aux=read(0x03);
    aux=read(0x04);
    aux=read(0x05);
    aux=read(0x06);

    //Registers settings for frame capture
    write(0x10,0x00);//Config_2 reg, set rest_bit to 0
    write(0x12,0x83);//set Frame_capture register
    write(0x12,0xC5);//set Frame_capture register
    wait_ms(20);//delay to store a complete frame.

    //Read a frame
    ncs=0;//able spi
    spi.write(0x64);//send pixel_burst address
    wait_us(160);

    //Save all pixels
    for(i=0;i<n_pixel;i++){
        frame[i]=spi.write(0x00);//read pixel_burst register
        wait_us(15);//delay between 2 successive reads.
    }
    ncs=1;//disable spi
```

```
wait_us(4);

//Power and reset sequence.
nreset=1;
ncs=1;
ncs=0;
write(0x3A,0x5A);//Power up reset register
wait_ms(50);
aux=read(0x02);
aux=read(0x03);
aux=read(0x04);
aux=read(0x05);
aux=read(0x06);

//READ squal, pixel's features, shutter.
squal=read(0x07);
av_pixel=read(0x08);
max_pixel=read(0x09);
min_pixel=read(0x0A);
shutter_up=read(0x0C);
shutter_low=read(0x0B);

//SEND data to host pc.
for(i=0;i<n_pixel;i++){
    pc.printf("%d",frame[i]);
    pc.printf("\t");
}
pc.printf("%d",squal);
pc.printf("\t");
pc.printf("%d",shutter_up);
pc.printf("\t");
pc.printf("%d",shutter_low);
pc.printf("\t");
pc.printf("%d",av_pixel);
pc.printf("\t");
pc.printf("%d",max_pixel);
pc.printf("\t");
pc.printf("%d",min_pixel);
pc.printf("\n");//termination char.
```

```
    wait_us(delay);//DELAY of cycle.
}//end while

}//end main

void write(int address, int data){

ncs=0;
spi.write(address+128);//put msb=1 on register address.
spi.write(data);//send data.
wait_us(35);
ncs=1;
wait_us(180);
}

uint8_t read(int address){

uint8_t out;
ncs=0;
spi.write(address);//put msb=0 on register address.
wait_us(160);
out=spi.write(0x00);//read data.
wait_us(1);
ncs=1;
wait_us(20);
return out;
}
```

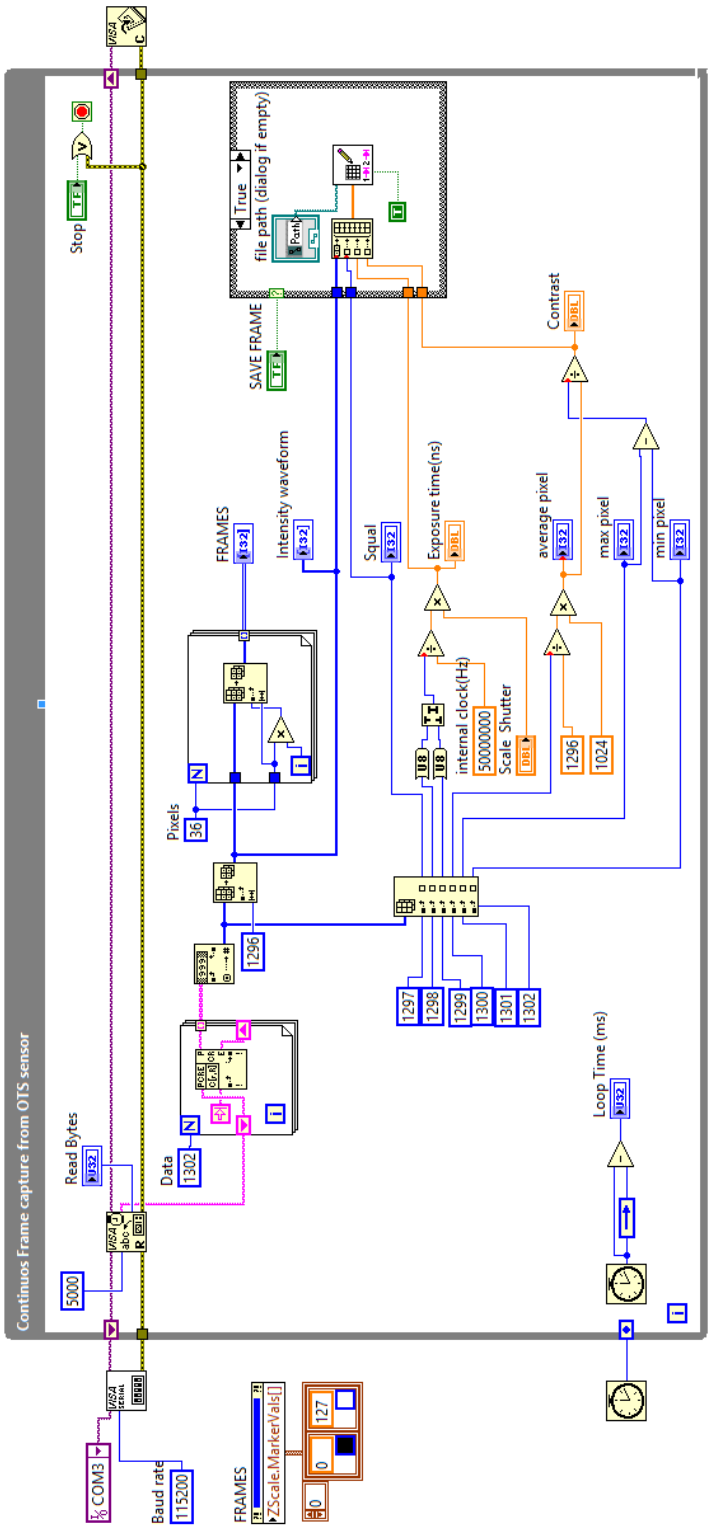


Figure 4.8: Frame acquisition program: block diagram

Chapter 5

System calibration

This chapter contains the experimental results for calibrating the system. Those tests has been performed in the Optical Measurements Laboratory from Politecnico di Milano. The first part is focused in several experiments in order to choose the best sensor and to understand the sensor limits. In the second part, was proposed a 2D mapping method to choose the optimal position of the sensor over the piston rod. The third part focus on the image acquisition for calibrate the pipe application. Finally, it was developed a system able to measure the linear and angular position on the piston rod.

5.1 Calibration setup

A general description of the hydraulic cylinder under test, focusing on external characteristics, has to be given to understand the encountered measurement during the calibration phase.

Technical Specifications	
Stroke	600mm
Working temperature	-30 to 80
Piston rod material	Chromium-plated
Piston rod wrinkledness	Ra Max 0.8um
Piston rod diameter	30mm
Cylinder material	Iron(Fe)

Table 5.1: Cylinder specifications

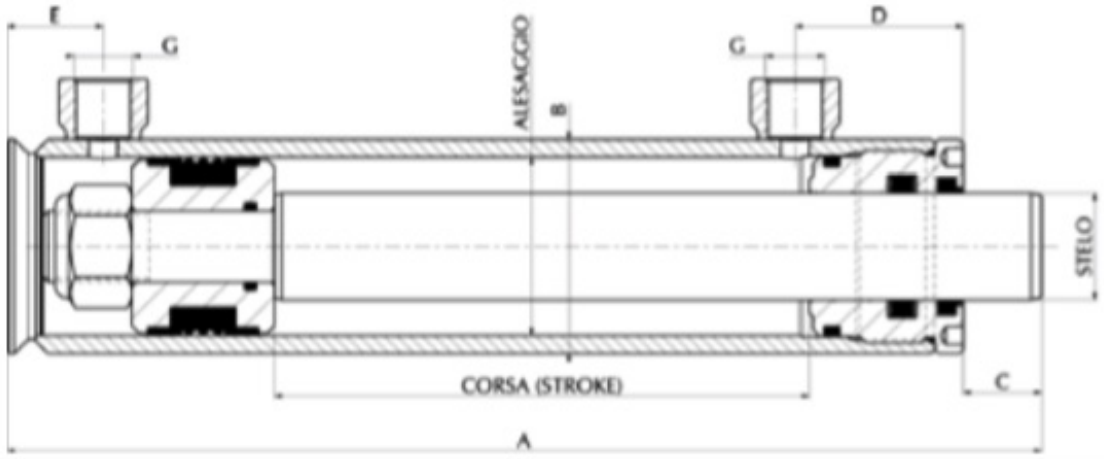


Figure 5.1: Hydraulic Cylinder with stroke=600mm

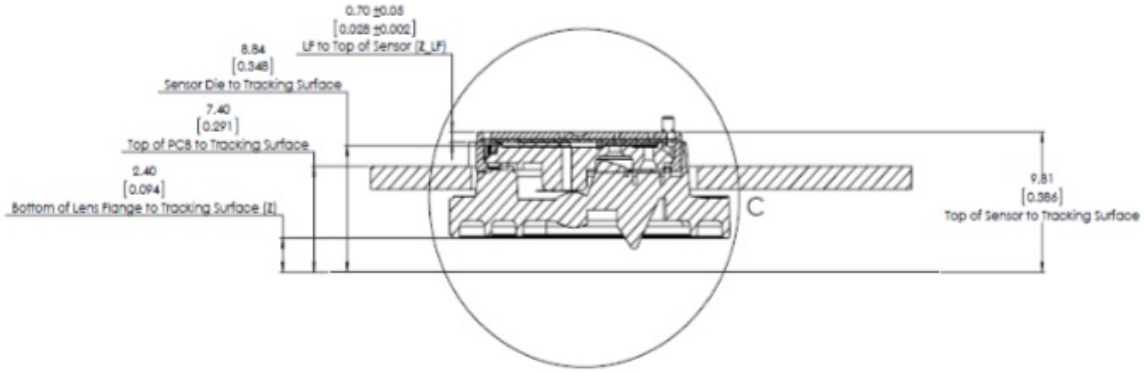


Figure 5.2: OTS and LGS sensor's section

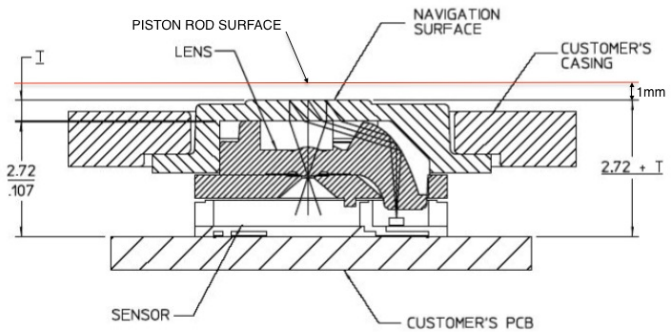


Figure 5.3: OFN sensor's section

In the calibration setup is important to consider the distance between the sensor's lens and the tracking surface, thus the piston rod in our case. In figures (4.2) and (4.3) we show the sensor's sections. The sensors has to the positioned with a distance from the tracking surface of 2.4mm with a tolerance of +/- 0.2mm for OTS and LGS sensors and with a distance of 1mm for the OFN sensor. This fixed distance stands for the lens-surface distance that a CMOS sensor needs for being around his focal length and so to give a good measure.

The calibration setup for our experiments is the following:

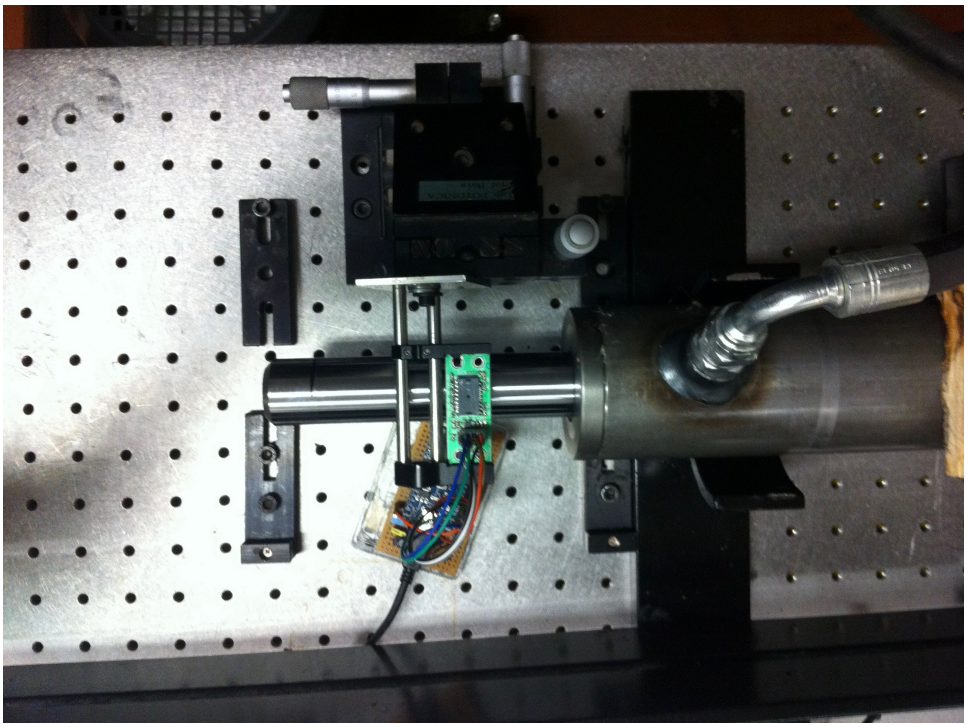


Figure 5.4: Calibration setup

The piston rod surface has some characteristics that has to be considered. The most important are:

- is not a plane, but convex. Due to the lateral illumination we can find some problems when positioning the sensor.
- is chromium-plated, and is reflective. So when the surface is illuminated the pixel's frame can saturate, lowering the contrast, obtaining a bad measurement.

5.2 Experiments

A first experiment was done to choose the best sensor for position sensing. Then, a second experiment aims to see the dependance of displacement with the sample period. Then, a third experiment aims to see the variation of sensitivity in function of the piston rod velocity. Finally, a fourth experiment is to verify if the sensor is able to detect laser markers over the piston rod.

5.2.1 Experiment 1

To choose the best sensor for position sensing, was acquired five back and forth displacements over the piston rod with a sampling period of 100ms. The measurements were done for three sensors under test, thus OTS, OFN and LGS sensors. The measurement by each sensor was acquired in parallel to another displacement measurement system used as groundtruth or reference.

In this experiment, a Laser displacement sensor(LDS) was used as a reference to measure displacement in an stable way. A CCD laser sensor (Keyence LK-G152) with a maximum sampling rate of 50 Hz was used in this study. This model has a built-in processor function such that the displacement data are outputted while power is supplied according to the preset data acquisition period. Thus, the distance between the LDS and the piston rod can be determined without a separate reader.

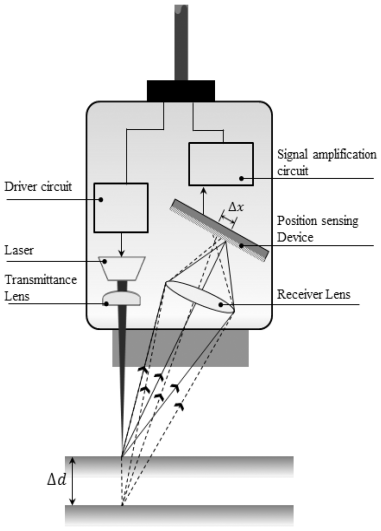


Figure 5.5: LDS principle

As depicted in Figure (4.3), the measurement principle of the optical LDS technique is that a laser beam, often with a diameter on the order of millimeters, is scattered when the target is reached, and this scattered beam creates an image on a one dimensional position-sensing device that is then converted into an electrical signal. The distance between the LDS and target can be triangulated from the positional information of the imaged laser beam.

As we briefly description in the last chapter, the OFN Sensor is a finger navigation sensor and has a particular characteristic to become a good candidate for solving the problem of the working temperature. This feature is important when the hydraulic cylinder works for several hours in working environments that can't help sensors to works well. As depicted in figure (4.5), comparing the sensor with the reference we can realized that the measure has a drifting in time because it integrates an error at each sampling period.

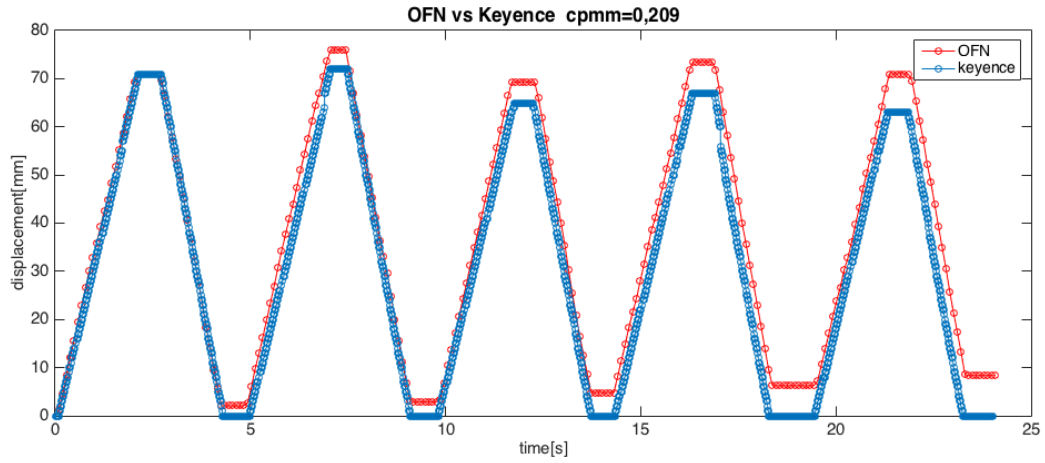


Figure 5.6: OFN sensor vs Keyence

The LGS Sensor is a laser based sensor, so we expect a better measurement than previous one. As depicted in figure(4.6), comparing the sensor with the reference we can realized that it the LGS follows very well the reference for all the time that measurement is done. In addition, it also shows a drifting, but is lower than previous one.

Finally, the OTS Sensor is sensor used for tracking purpouses, so we expect also good results. As depicted in figure(4.7), comparing the sensor with the reference we can realized that the OTS follows very well the reference for all the measurement time. In addition, it also shows a drifting, but is lower than OFN.

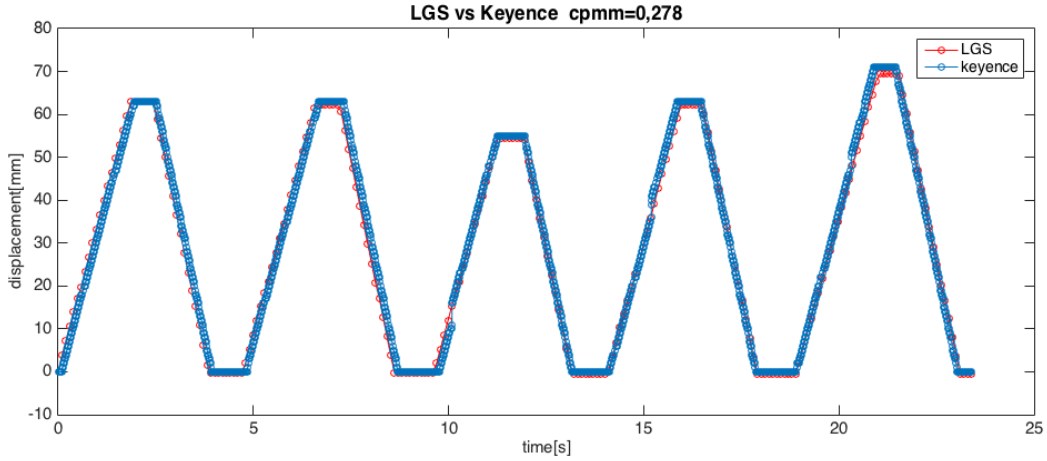


Figure 5.7: LGS sensor vs Keyence

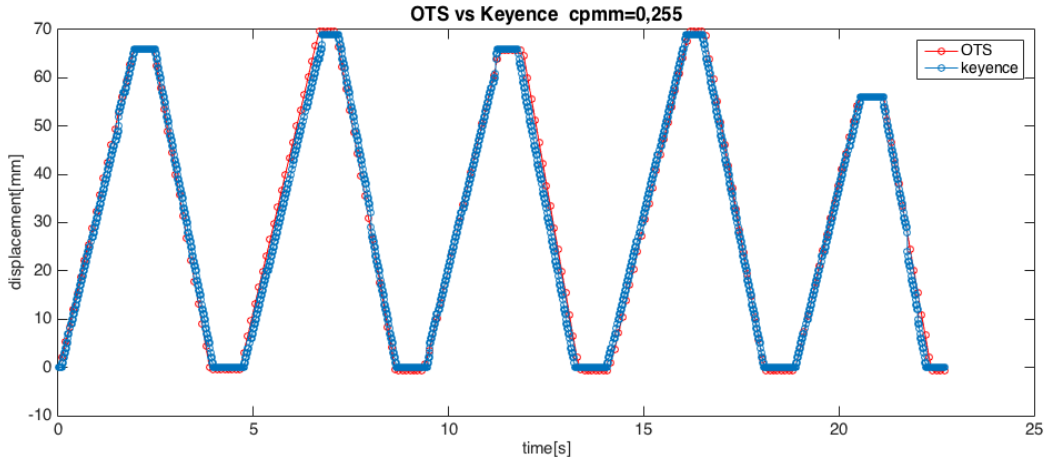


Figure 5.8: OTS sensor vs Keyence

Conclusion Nonetheless the OFN sensor was the best candidate to fulfill all our requests, it couldn't give a relying measurement because its drifting in time is the worst between the three sensors under test. The OTS and LGS sensors has similar performances, but considering that OTS sensor is a led based sensor,so consumes less power, and is a sensor used for tracking purposes it has become the best candidate for our measurement system.

5.2.2 Experiment 2

In this experiment we wanted to investigate the stability of the optical sensor chose in the Experiment 1, thus the OTS sensor. The stability is investigated over a range of different sampling periods with the aim to show how the drifting in time varies with different sampling periods of 20ms, 100ms, 250ms, 500ms and 1s.

The experiment was done acquiring five back and forth displacements in terms of counts over the piston rod. We expect that doing a back and forth displacement the measure returns to zero if the sensor gives a good measure.

In the depicted figure (4.7) we can see the results of the back and forth measurements using a sampling period of 20ms. In addition, in the figure (4.8) we can see how the drifting evolves in time during the back and forth measurements.

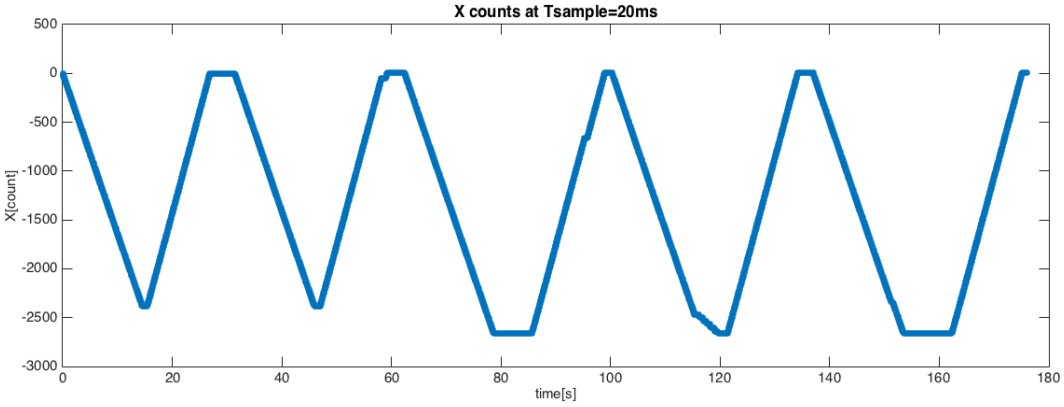


Figure 5.9: X counts with a sampling period of 20ms

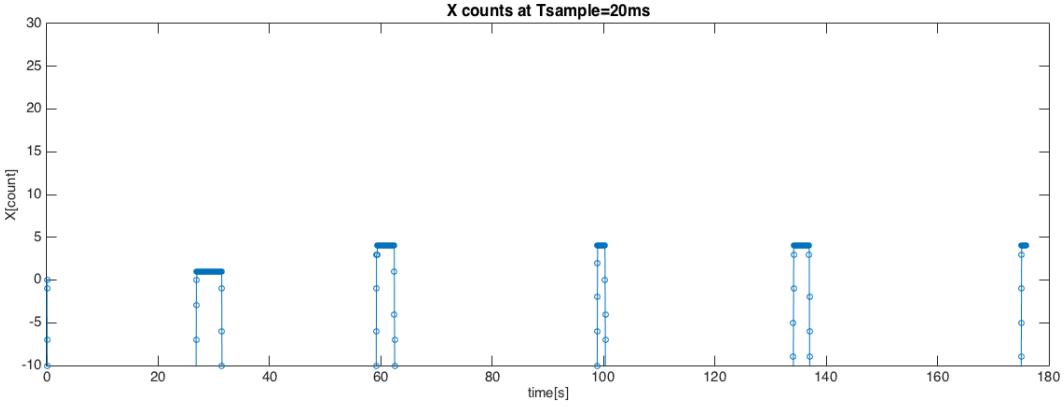


Figure 5.10: Drift evolution with a sampling period of 20ms

In the depicted figure (4.9) we can see the results of the back and forth measurements using a sampling period of 100ms. In addition, in the figure (4.10) we can see how the drifting evolves in time during the back and forth measurements.

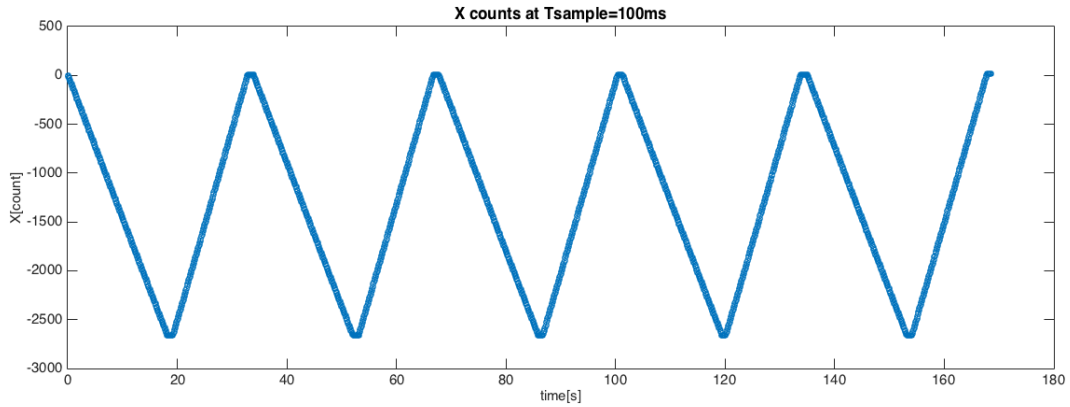


Figure 5.11: X counts with a sampling period of 100ms

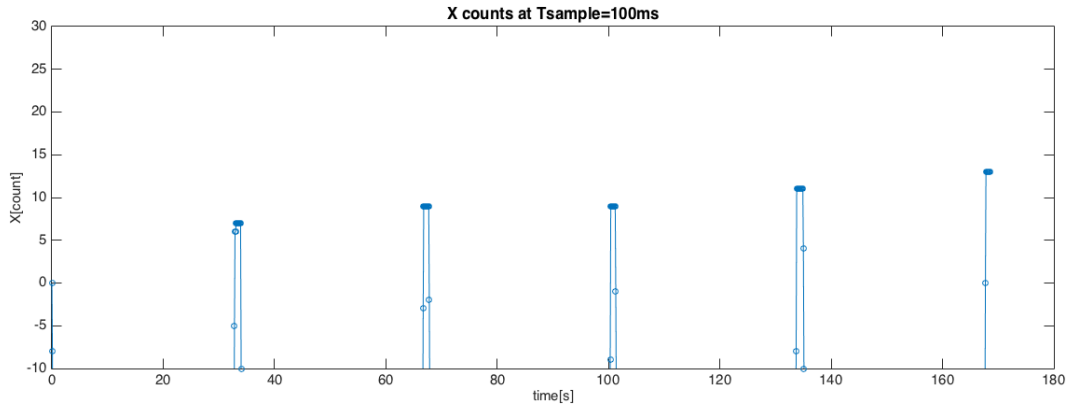


Figure 5.12: Drift evolution with a sampling period of 100ms

In the depicted figure (4.10) we can see the results of the back and forth measurements using a sampling period of 250ms. In addition, in the figure (4.11) we can see how the drifting evolves in time during the back and forth measurements.

In the depicted figure (4.12) we can see the results of the back and forth measurements using a sampling period of 500ms. In addition, in the figure (4.13) we can see how the drifting evolves in time during the back and forth measurements.

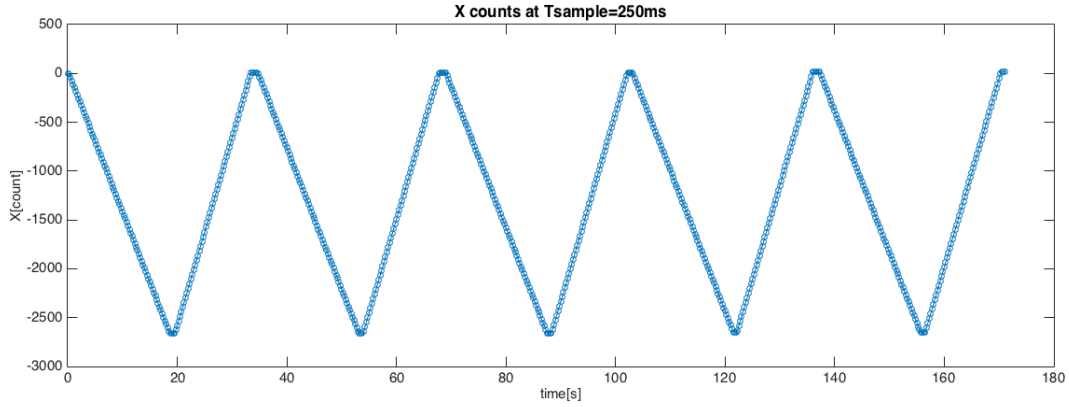


Figure 5.13: X counts with a sampling period of 250ms

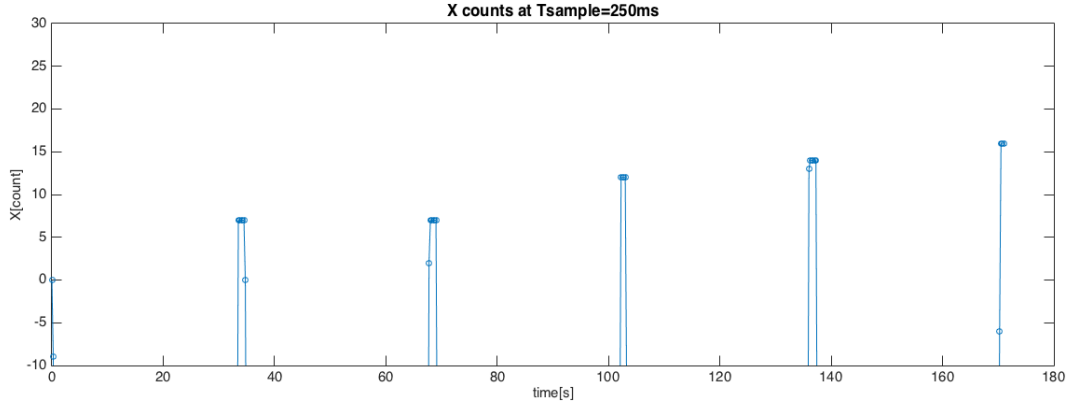


Figure 5.14: Drift evolution with a sampling period of 250ms

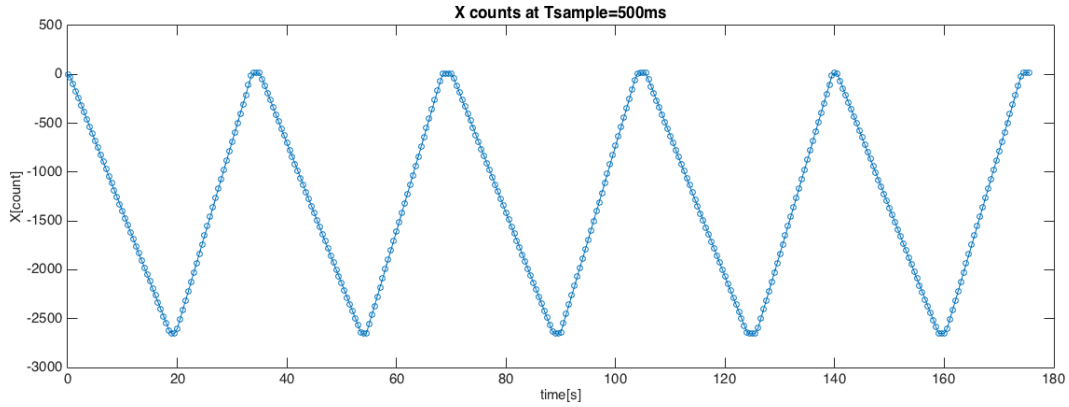


Figure 5.15: X counts with a sampling period of 500ms

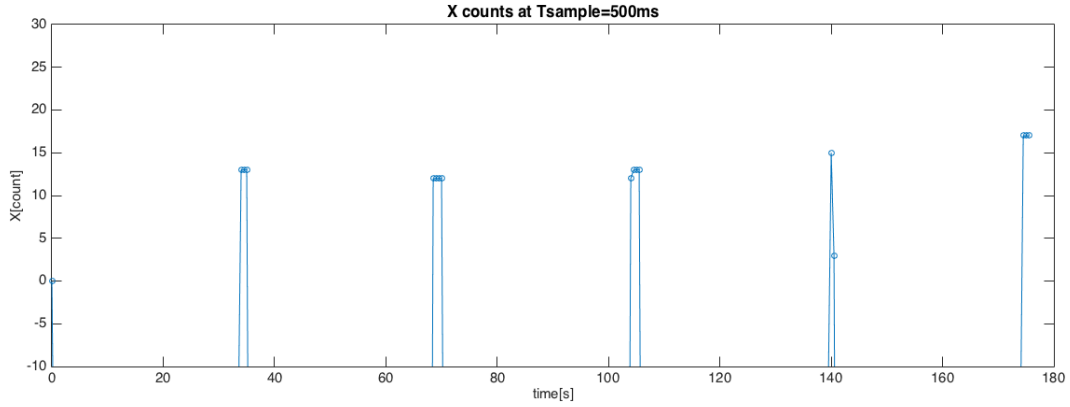


Figure 5.16: Drift evolution with a sampling period of 500ms

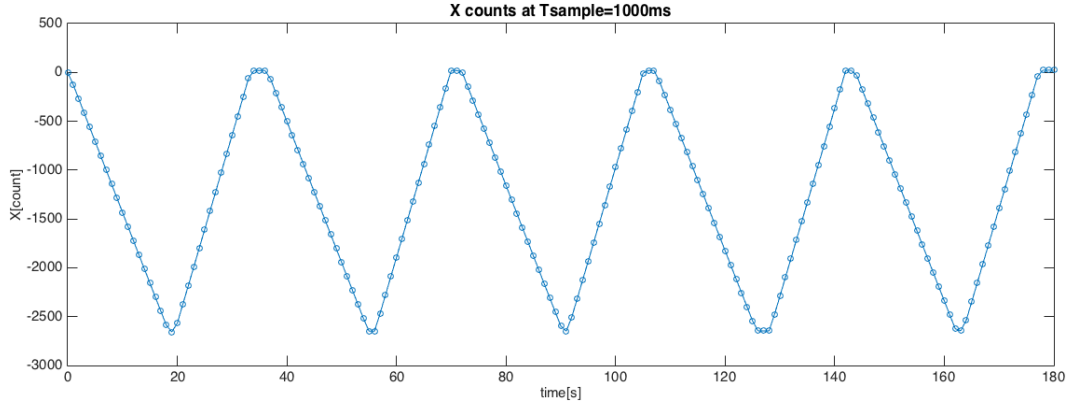


Figure 5.17: X counts with sampling period of 1s

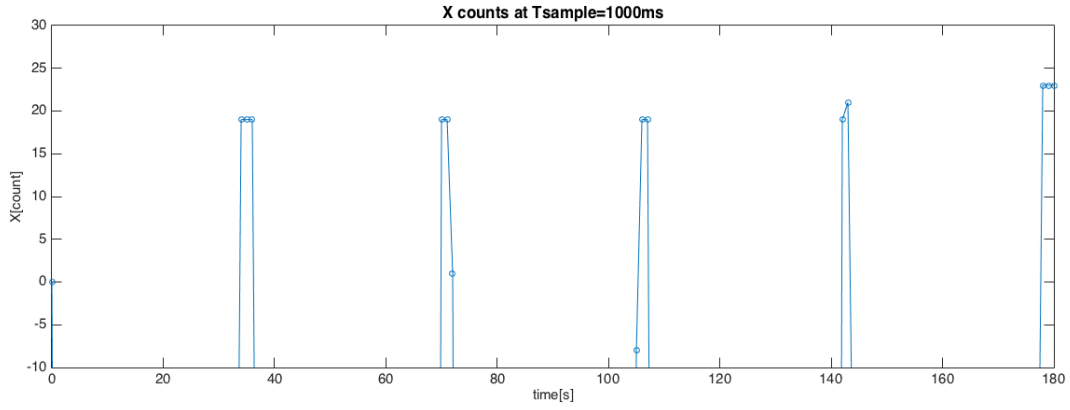


Figure 5.18: Drift evolution with a sampling period of 1s

Finally in the depicted figure (4.14) we can see the results of the back and forth measurements using a sampling period of 1s. In addition, in the figure (4.15) we can see how the drifting evolves in time during the back and forth measurements.

Conclusions As we see the drifting evolution in time is worse for high sampling periods.

5.2.3 Experiment 3

In this experiment we want to see if different speeds varies the effective sensitivity of the sensor that stands for read counts per millimeter (cpmm). This experiment was done because during the experiments we realized that back and forth movement phases had different velocities. The velocity in our hydraulic cylinder is determinate by the motor that allows the passage of oil from the reservoir to the double acting cylinder. To obtain different speeds we need different flows of oil, but we don't have a complete control of this parameter. As a consequence back and forth movement phases travels at different speeds and in this way, for a fixed stroke, we obtain different number of counts.

Initially, we suppose that the back speed was greater than the forth one and its consequence is a negative drifting of the measure in time. Now if we try to slow down the back speed manually, controlling the jostick that drives the oil flow, we can obtain a back speed lower than in the forth phase. We expected that in this case the drifting of the measurement becomes positive. We can see the results that confirm our thesis. The experiment was done acquiring four back and forth displacement along a stroke of 600mm with a sampling period of 100ms.

Conclusions So now, we can confirm that the tracking speed is direct proportional to the read number of counts by the sensor and in the same way to the sensitivity. We expected this because if the speed increases the sensor see the images wider than before. It is a problem related to its shutter speed or exposure time. With this information we can develop a an algorithm to compensate or lower the error of the displacement measurement.

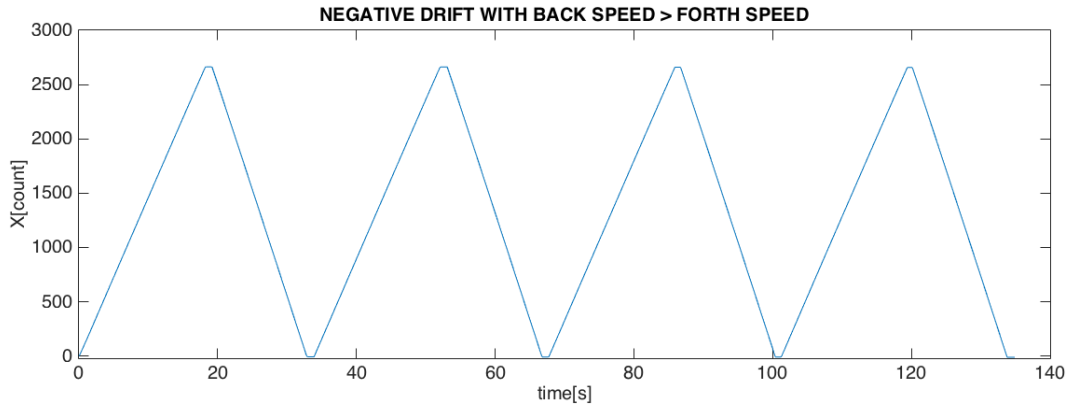


Figure 5.19: X counts with negative drift

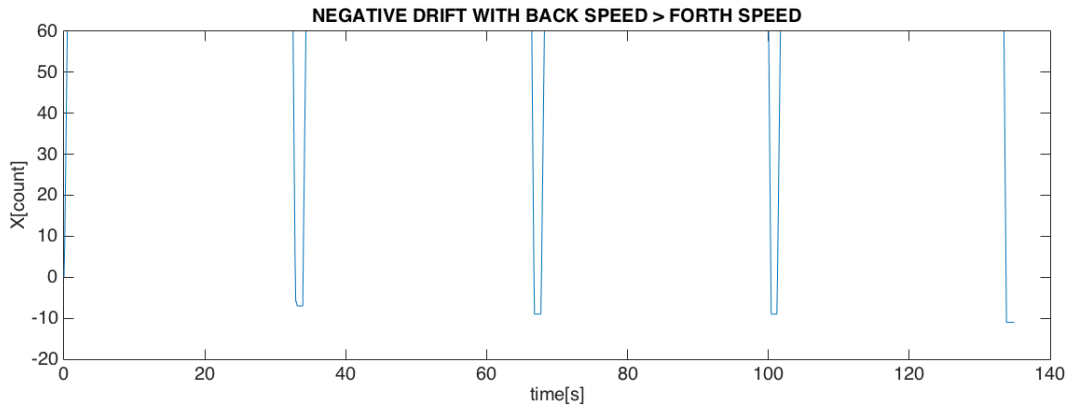


Figure 5.20: Negative Drift evolution for 4 back and forth measurements

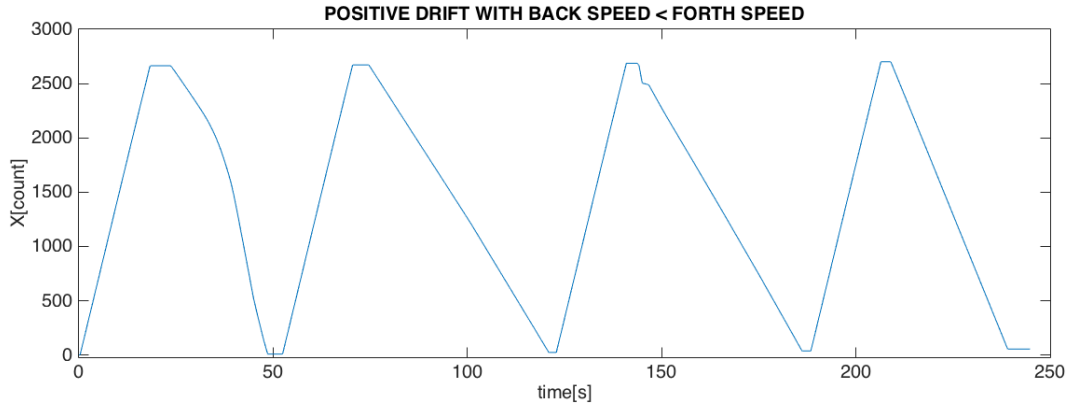


Figure 5.21: X counts with positive drift

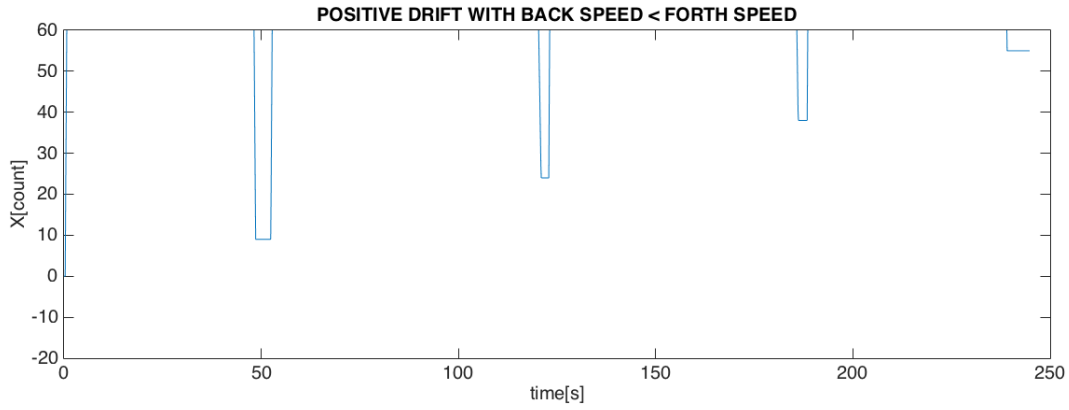


Figure 5.22: Positive Drift evolution for 4 back and forth measurements

5.2.4 Experiment 4

With this experiment we want to find and alternative solution for a possible algorithm proposed in the Experiment 3. Using absolut references in the working environment we can compensate the displacement measurement lowering the integrated error accumulated over time. With this idea were done two experiments with sampling periods of 20ms and 100ms, in order to understand if the sensor is able to detect a marker along the piston rod. The marker is done with a black enamel and is positioned at a half of the entire stroke. In addition, we also want to control what sensor parameter between squal and shutter give the best feedback in order to detect the marker under test.

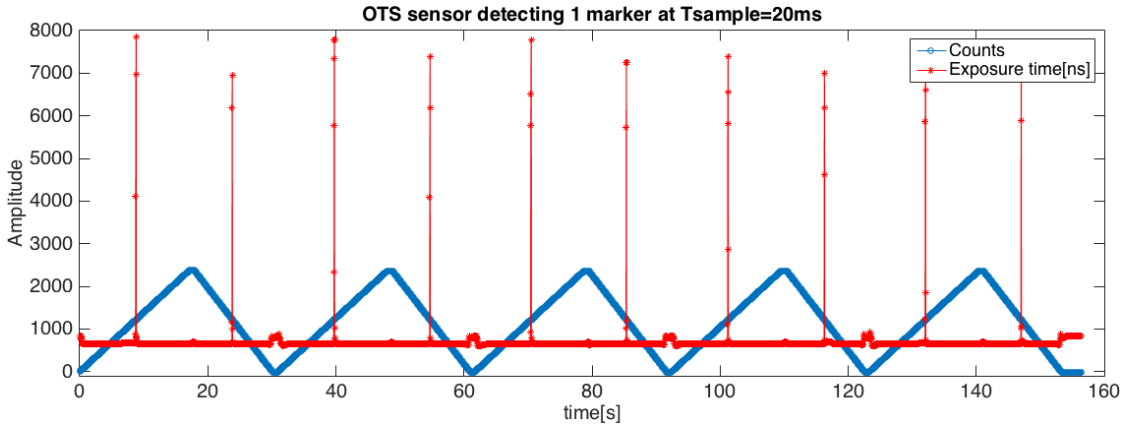


Figure 5.23: Marker detection at 20ms

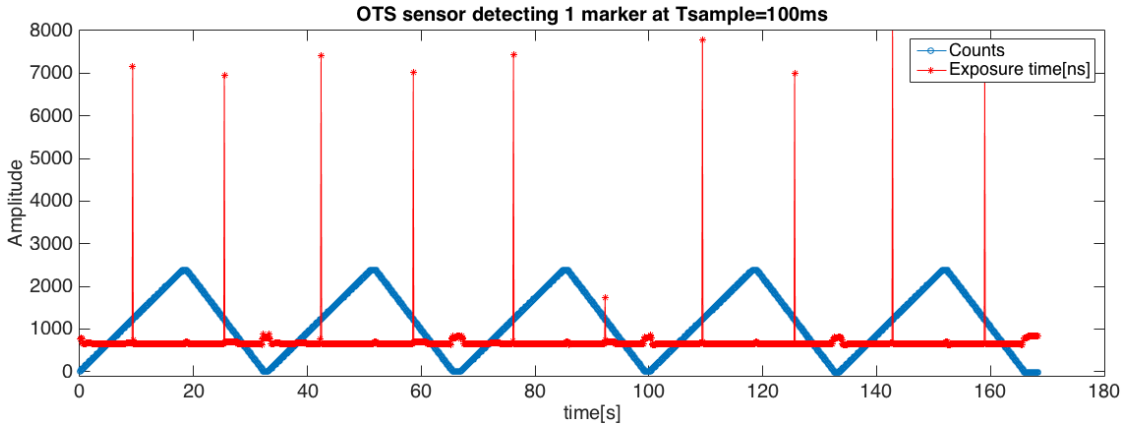


Figure 5.24: Marker detection at 100ms

Conclusion The experiment gives back good results. Looking for the best parameter that gives the feedback to detect the marker we choose the shutter speed or exposure time because it gives a high variation of its intensity when marker pass under the sensor. Instead, the squal value gave a too uniform value during the marker passage, so we consider not reliable for our purposes.

Relying on this results, we can develop an algorithm that will be presented in the next chapter.

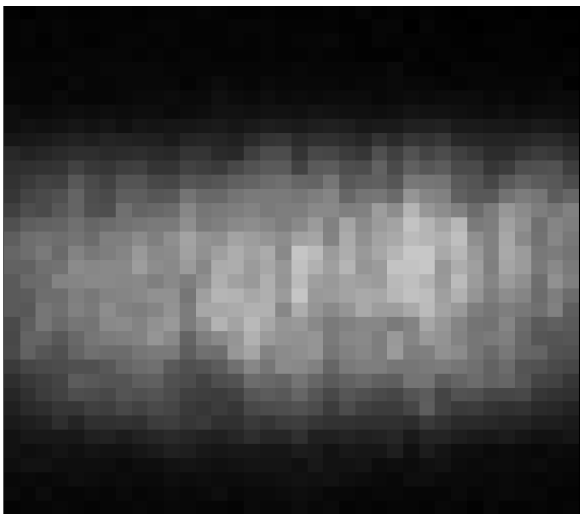


Figure 5.25: Surface image in a piston rod with diameter of 20mm

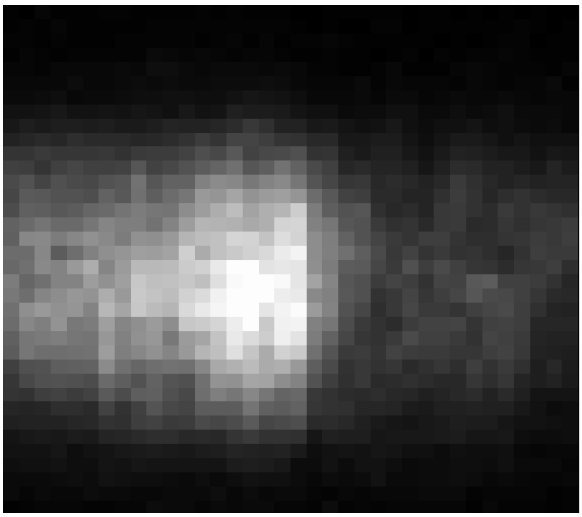


Figure 5.26: Marker captured by the sensor

5.3 2D Mapping

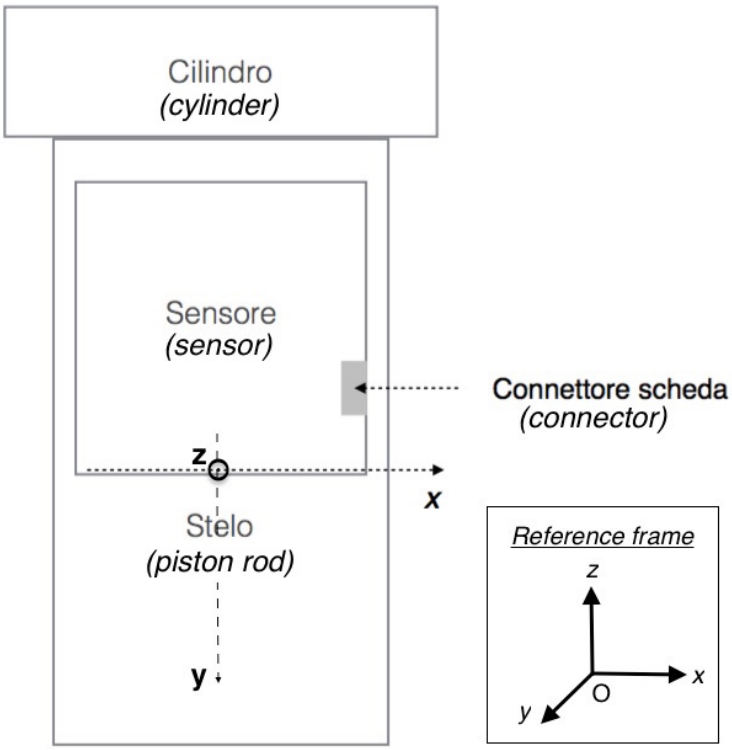


Figure 5.27: Setup 2D mapping

This method aims to find the optimal position of the OTS sensor over the piston rod. It can be done searching experimentally the best position, in an orthogonal frame reference to the piston rod, where the displacement measure is visible and not quantized.So, the best position can be found considering displacement variations over the x axis and the z axis. In this way can be defined an specific area where the measurement is retained valid.

The tests were done over hydraulic cylinders with piston rods for different diameters: 20cm, 30cm and 40cm. The chose to test different diameters is done because we expected to find valid areas proportional to diameter dimensions. This is because the surface acquired by the sensor is proportional to the piston rod diameter.

Piston rod with D=20cm Over the x axis, the sensor has valid measurements between +1.1 and +3.6 mm. Between -1.1 and -2.1 is detected a further valid range but the error tollerance present in the negative range is lower than in the positive range. Over the z axis, the sensor has valid measurements from 8.81mm to 10.81mm.

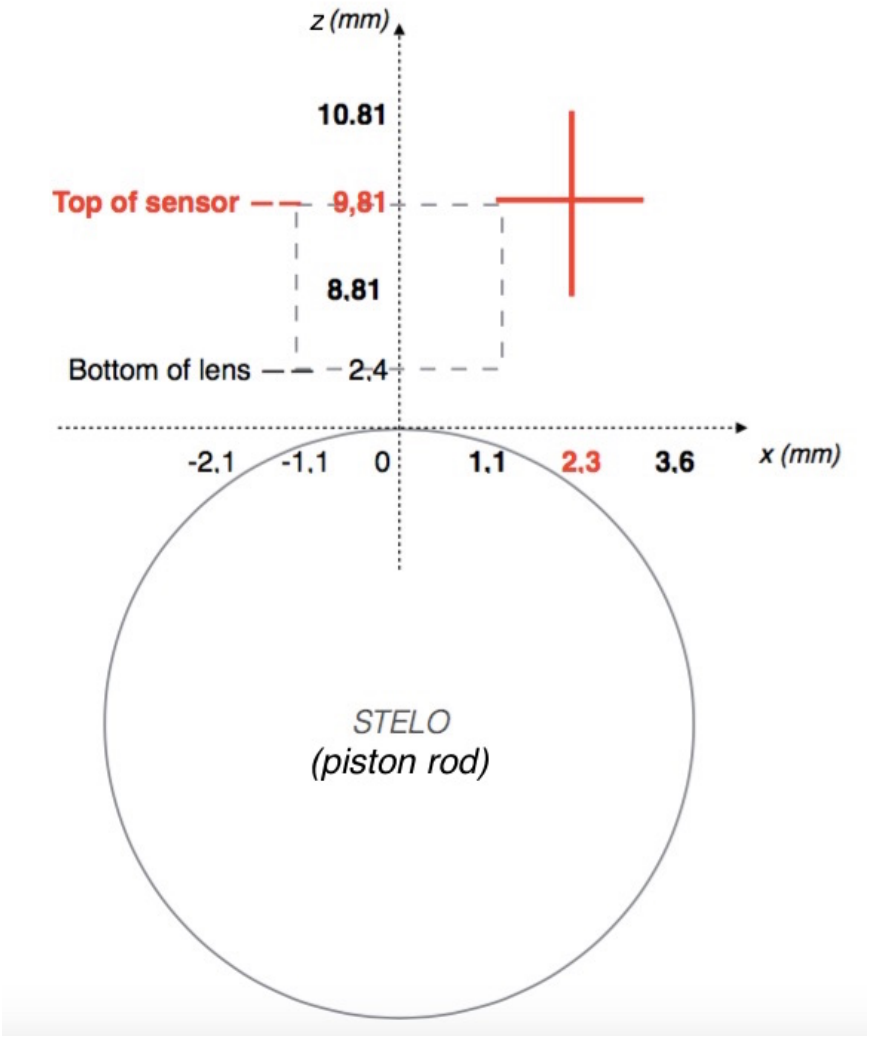


Figure 5.28: 2D Mapping for Rod D=20mm.

Piston rod with D=30cm Over the x axis, the sensor has valid measurements between 0 and +3.5 mm. Between 0 and -0.25 is detected a further valid range but the error tollerance present in the negative is lower than in the positive range. Over the z axis, the sensor has valid measurements from 9.31 mm to 10.81 mm.

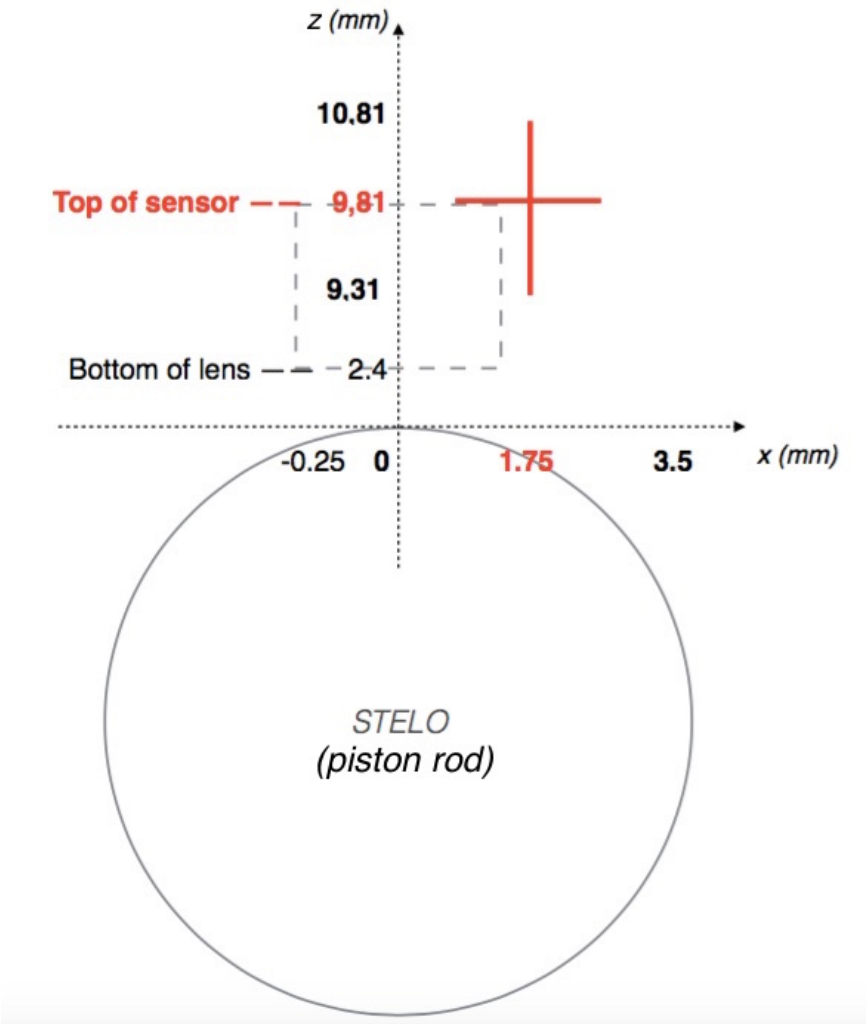


Figure 5.29: 2D Mapping for Rod D=30mm.

Piston rod with D=40cm Over the x axis, the sensor has valid measurements between 0 and +6 mm. Between [0, -1] and [-1.1, -2.1] is detected a further valid range but the error tollerance present in the negative range is lower than in the positive range. Over the z axis, the sensor has valid measurements from 8.81mm to 11.31mm.

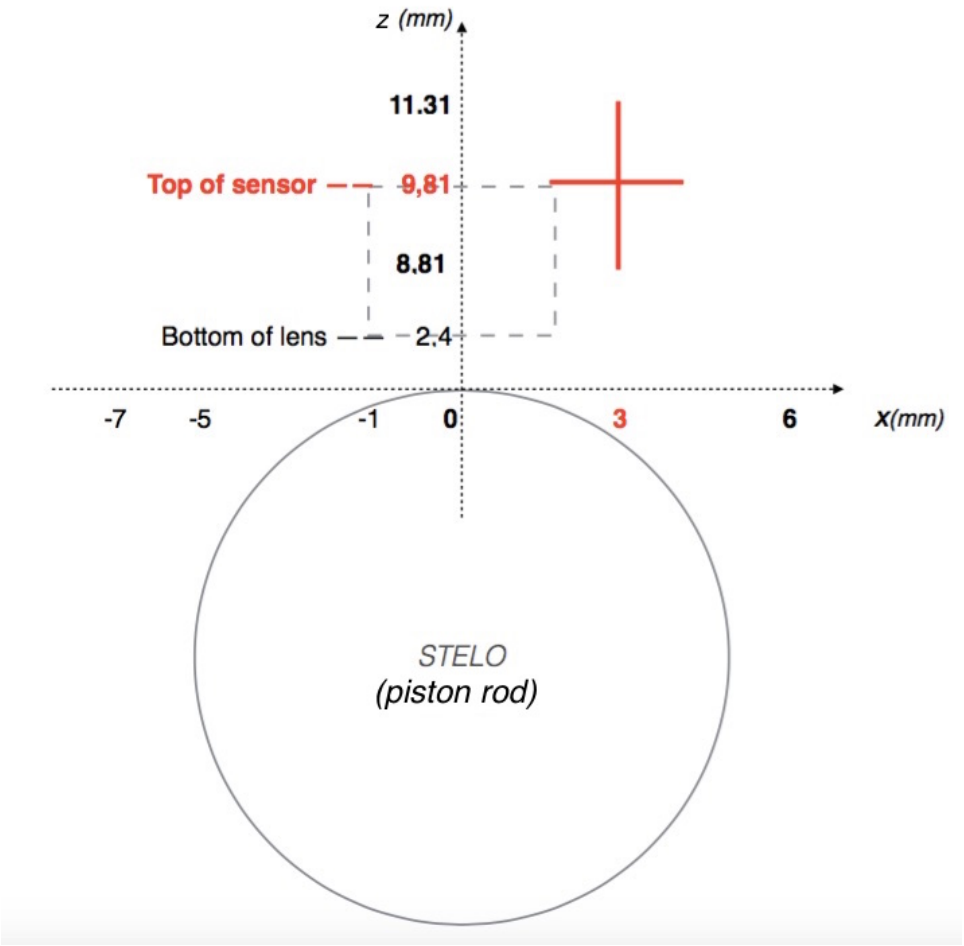


Figure 5.30: 2D Mapping for Rod D=40mm.

Conclusions As we expected for the a piston rod with a bigger radius we have more error tolerance. Successively was developed an ABS support with a 3D printer used in the final part of the experiments. Moreover, was develop a mechanical part that will be inserted on the cylinder head or gland and so it will be integrate in the hydraulic cylinder.

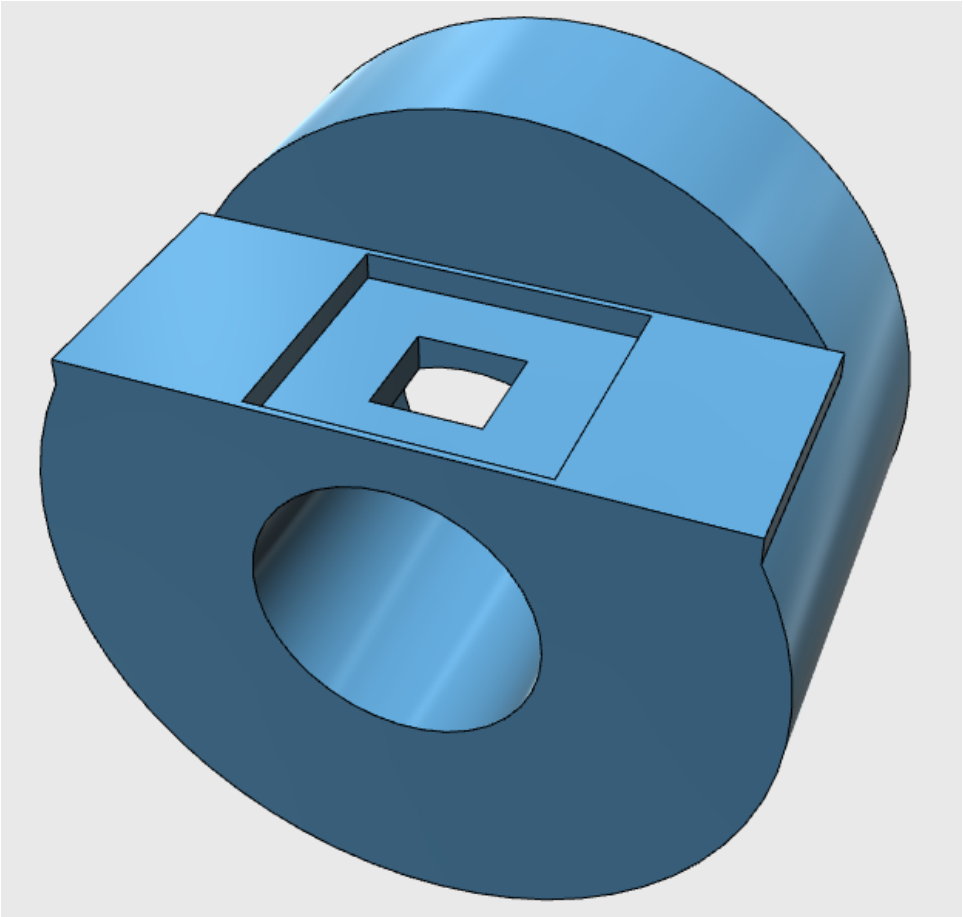


Figure 5.31: ABS support for OTS sensor

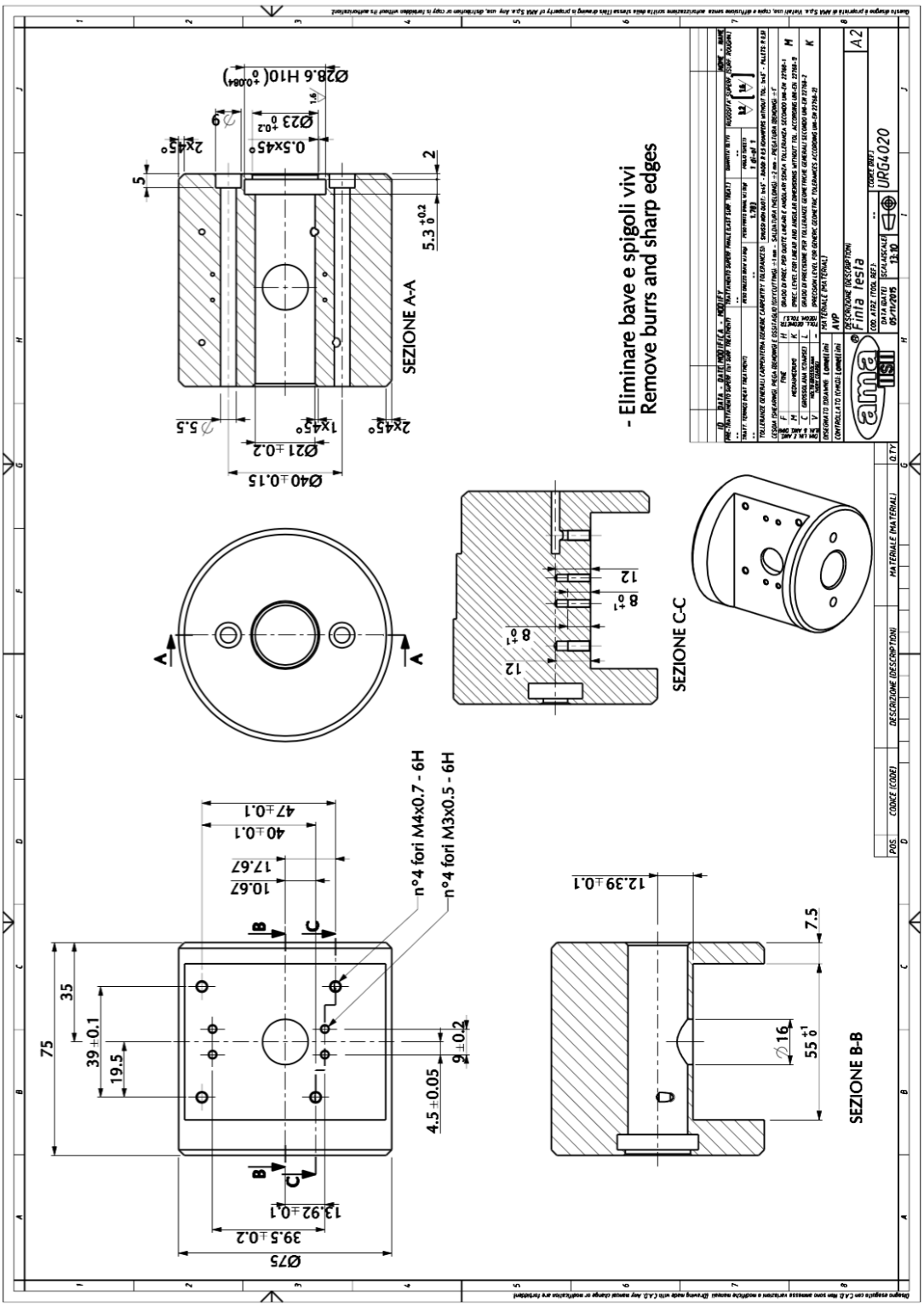


Figure 5.32: Mechanical support for cylinder's head

5.4 Pipe system calibration

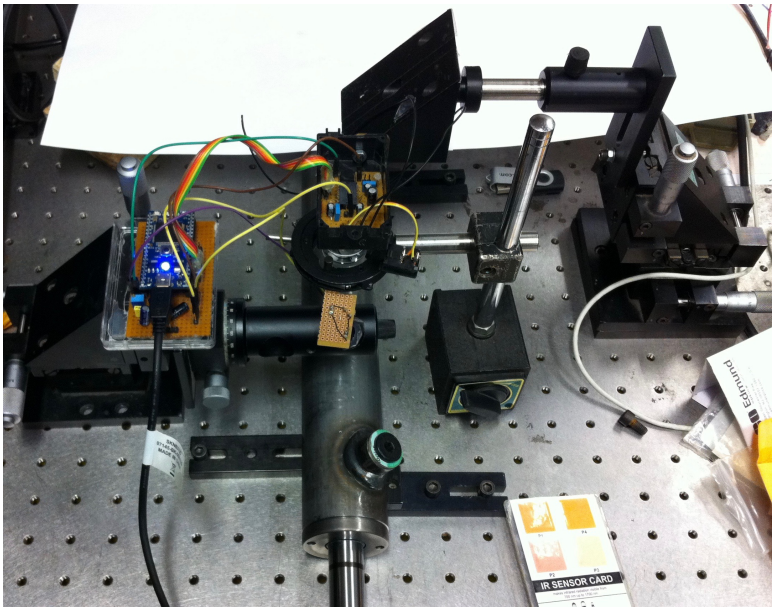


Figure 5.33: Pipe system setup

The pipe system was considered for an alternative application of the hydraulic cylinder system. It aims to measure the pipe circumference from a 50mm of distance from the its surface. It can be done only changing the optics of the mouse sensor, because its own optics lenses are design to allow to detect measurements, from a tracking surface, with a height of 2.5mm and a tolerance of +/- 0.5 mm.

For this experiment, were used three double-convex lenses with different focal lenses. The lenses specifications will be summarized in the following table:

Lens Specifications			
Features	Lens 1	Lens 2	Lens 3
focal length(mm)	13.5	16	25.4
diamater(mm)	9	16	18
S2(mm)	18.5	23.53	51.62
FoV	4.24	3.6	2.24

Table 5.2: Lens specifications

To do a good calibration of the camera was useful to use a checkerboard. It was used with the aim to correct eventually blurs produced when looking for the optimal distance lens-pinhole camera. The checkerboard has 5x5 pixel for each square and was printed with a 1200 dpi resolution.

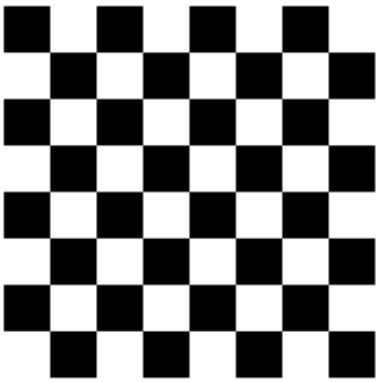


Figure 5.34: Checkerboard used for camera calibration

For the illumination system was choose a SFH-480 LED with a wavelength of 880nm and a maximum power of 200mW. In the followings figures will be shown the images captured with the different lenses:

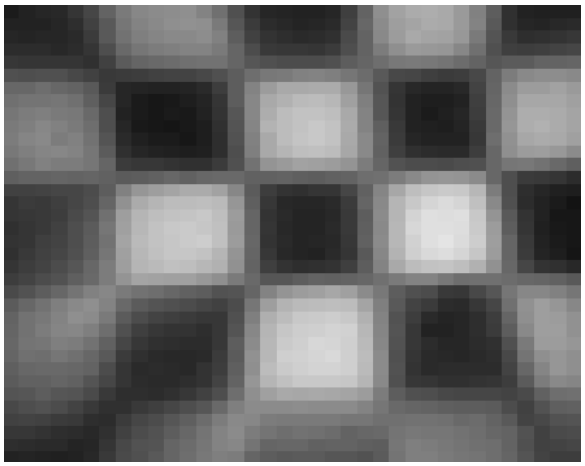


Figure 5.35: Image captured using a lens with focal length=13.5 mm

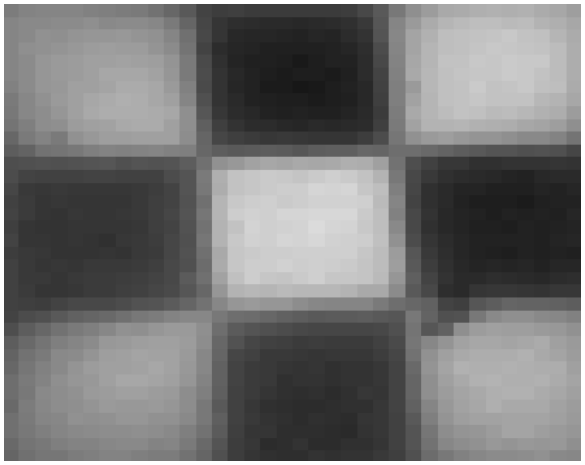


Figure 5.36: Image captured using a lens with focal length=16 mm

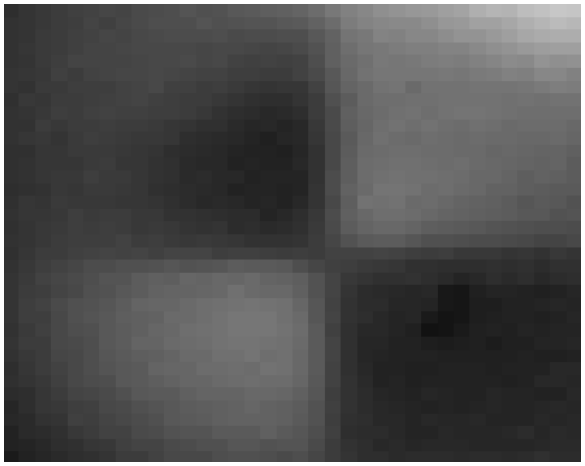


Figure 5.37: Image captured using a lens with focal length=25.4 mm

As we can see from the images, due to the convex surface, the images are sharper on the central regions and blurred on the lateral regions. We expected that, because the distances from the camera to the surface points are different because the surface is not a plane. So this fact can produce a blur image and so the DIC algorithm can fails when measuring the displacement.

Chapter 6

Measurements

In this chapter will be presented the algorithms proposed in the previous chapter, with his respective results. First of all, will be presented an algorithm that do a direct compensation of the read counts using a single absolute reference along the piston rod. Then, will be presented a second algorithm that take in account two absolute references instead of one. Finally, will be proposed and alternative compensation method knowing that read counts, for a fixed stroke, varies with the tracking speed.

6.1 Marker Algorithms

In the previous chapter, precisely in the Experiment 4, we demonstrate that the shutter speed or exposure time of the sensor varies his amplitude when the surface change his luminosity. In our case, the shutter value remains around a mean value of 700 ns, when the sensor tracks the piston rod surface, that is glossy. And the shutter value increase up to 1500 ns when the sensor detects the laser marker, that is opaque.

So, we can use the shutter variable to make a compensation of the displacement measure only when the sensor detects one or two markers. The marker is done by laser techniques, is long 3 mm and so sufficient for produce high shutter amplitude variances. The first marker is positioned, from the rod cap to the marker center, at mm, the second is positioned, rom the rod cap to the marker center, at xxx mm. All the markers are positioned along a piston rod with a stroke of xx mm.

In the following figure (6.1) we can have a detail of the exactly markers position in a piston with a stroke of xx mm and with a radius of 20 mm.

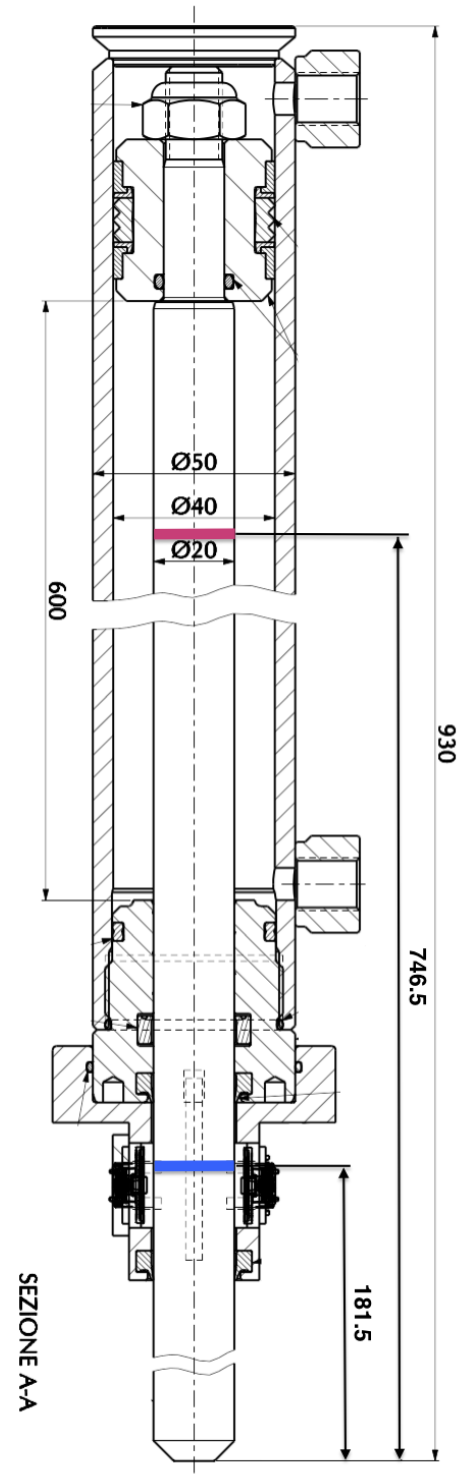


Figure 6.1: Cylinder with 2 markers

6.1.1 Algorithm 1

The first algorithm aims to do a compensation of the measure for the first marker positioned, from the piston rod's cap to the marker center, at 181.5 mm. So the first edge of the marker is at 180 mm (XoL), and the second edge is at 183 mm (XoH).

This algorithm is structured in the following way:

1. *Monitor the shutter speed or exposure time.* We have to monitor this variable at each cycle to monitor indirectly the surface luminosity.
2. *Control if the shutter exceeds a fixed threshold value.* The threshold value was decide in function of the average shutter value of the rod surface. That in our case is around 700 ns. So we fixed the threshold at 900 ns.
3. *Define a flag to do the compensation only one time.* The shutter signal is a rectangular waveform, with gradual edges and with a maximum when the sensor is just over the marker. The compensation has to be done only the first time that the sensor cross the shutter rising edge. So, defining a flag can be useful to control this. Initially, the flag is imposed at zero, and when the compensation is completed, its value change to one. The flag returns to zero only the next time the sensor cross the marker.
4. *Control delta to obtain the direction of movement.* We have to impose the first edge position(XOL) or the second edge position(XOH), to the OUT variable, depending on the direction of movement of the piston rod. The decision of impose XOL or XOH, can be done controlling the delta signal, that stands for the istantaneous displacement given by the sensor. This signal provides also the sign of the movement.
5. *Make the compensation.* At this stage, the compensation of the measure is done imposing the OUT value with a XoL value, that stands for the first edge position, or with a XoH value, that stands for the second edge position.
6. *Continue with the delta integration.* The delta integration, with its OUT value as result, continues to be done normally. It is done also without applying any algorithm.

In the depicted figure (6.2), will be shown the algorithm wrote using Labview. It was wrote as SubVI, and then recalled to the main program.

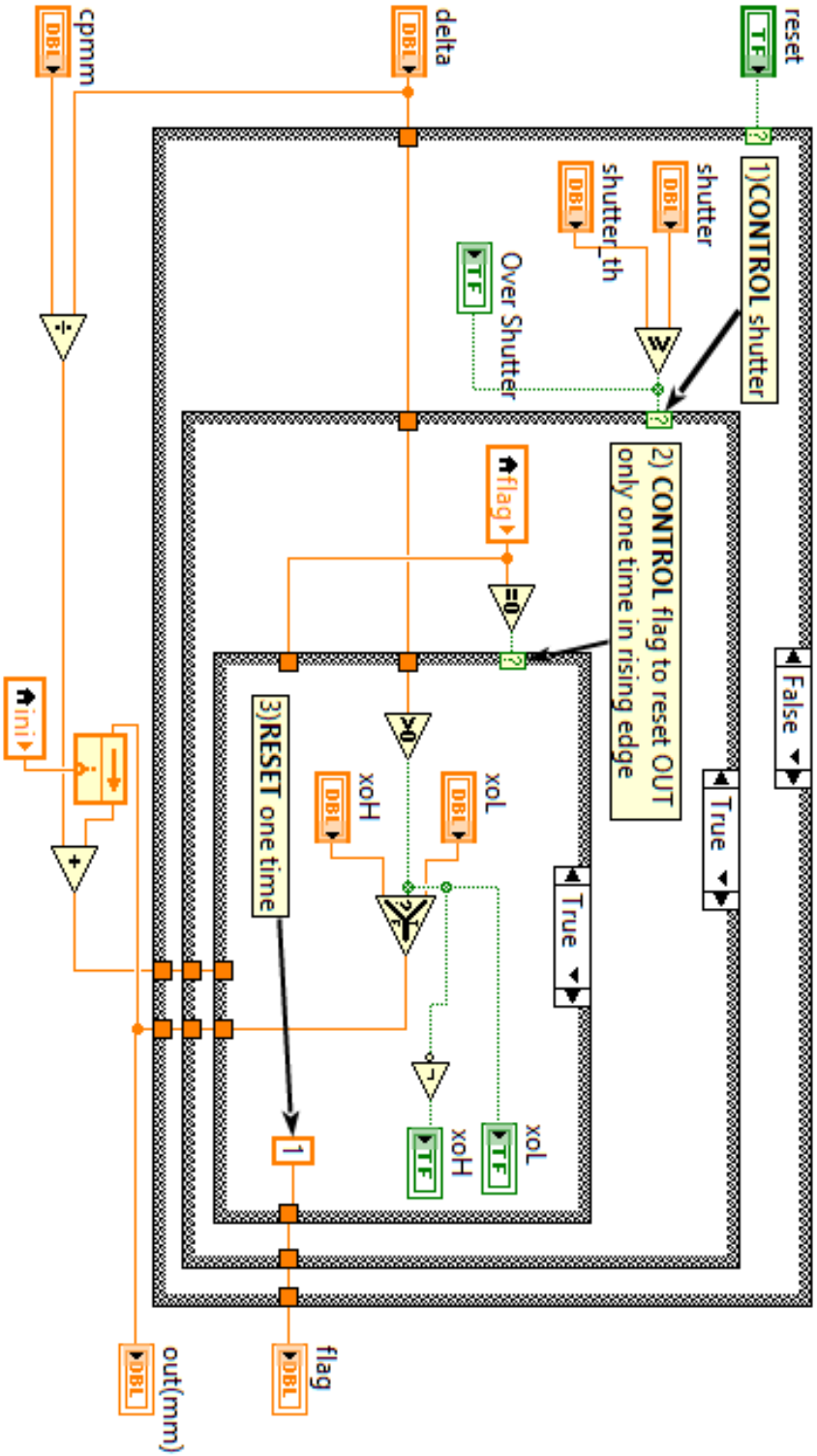


Figure 6.2: Algorithm for 1 marker

6.1.2 Algorithm 2

The second algorithm aims to do a compensation of the measure using two markers. The first marker is positioned, from the piston rod's cap to the marker center, at 181.5 mm. So the first edge of the first marker is at 180 mm (XoL1), and its second edge is at 183 mm (XoH1). The second marker is positioned, from the piston rod's cap to the marker center, at 746.5 mm. So the first edge of the second marker is at 180 mm (XoL2), and its second edge is at 183 mm (XoH2).

Will be proposed an algorithm that is able to detect two markers in order to a give feedback of the real piston position. It also has to detect marker's length in order to apply the compensation only for markers with 3 mm of length and not for other eventually markers. e.g. dusty markers.

First of all we have to define some key concepts, after explaining the algorithm structure:

1. *Define two stages of operation: Overshutter (OS) and Undershutter (US) conditions.* We defined this stages because in both condition we make operations that are useful for the correct algorithm operation.
    - OS condition: In this condition is done two important operations. The first one, makes the delta integration for measuring the exactly maker length, and the second one, makes the commutation of the marker state for deciding which marker we are going to use to compensate OUT value.
    - US condition: In this condition we control the marker state, marker length, direction of movement and make the compensation.
2. *Define three marker states.* We define three states to store in memory the marker that is going to compensate the OUT value, so the piston position. The states are the following:
    - zero state: It is a DEFAULT state. It is used for delta integration that increase or decrease OUT value.
    - first state: Is the state in relation with FIRST MARKER. In this state the compensation of OUT value with the first marker is done.
    - second state: Is the state in relation with SECOND MARKER. In this state the compensation of OUT value with the second marker is done.
3. *Define a second accumulator.* In addition, to the first integrator for the OUT value, so for the piston rod position, it is introduced a second integrator use

for measuring the marker length.

The Algorithm 2 is structured in the following way:

1. *Monitor the shutter speed or exposure time.* We have to monitor this variable at each cycle to monitor indirectly the surface luminosity.

2. *Control if the shutter exceeds a fixed threshold value.* The threshold value was decide in function of the average shutter value of the rod surface. That in our case is around 700 ns. So we fixed the threshold at 900 ns.

3. *Measure the marker length.* It's done with a second accumulator that integrates the delta, defined as the istantaneous displacement of the sensor. The measuring is done for all the time we are in OS condition, thus when the sensor is over the marker.

4. *Select the marker state.* To decide which marker chose, it was defined a fixed threshold position(TP). For the threshold position was chose the half position between the position of the first and the second marker. So its value is 464 mm. In order to select first or second marker state, was compared the OUT value, so the piston position, with the threshold position. If OUT is lower than TP, the marker state commutates to first state. Instead commutates to second state.

5. *Control the marker state.* Now we are in US condition. We have to control the marker state, previously decided with a defined criteria. In this stage we have to control which marker was chose for compensate the OUT value, so the piston position.

6. *Control if the marker length is in the desired range.* With the previous calculated marker length, we have to control if the marker belongs to a desired range. This range was defined to be sure that the compensation was applied also for markers with 3 mm of length and not for eventually other markers. Eg. dusty markers. In our case the valid range was defined between 2.5 mm (Marker lower) and 3.5 mm (Marker upper). So if the marker belongs to the range we can apply the compensation, otherwise the compensation fails.

7. *Control delta to obtain the direction of movement.* We have to impose the first edge position(XOL1/2) or the second edge position(XOH1/2), to the OUT variable, depending on the direction of movement of the piston rod. The decision of impose XOL1/2 or XOH1/2, can be done controlling the delta signal, that stands for the istantaneous displacement given by the sensor. This signal provides also the sign of the movement.

8. *Make the compensation only one time.* This phase is managed by the marker state. When the 5, 6, 7 phases were done, the maker state changes immediately to DEFAULT value and so the compensation is done only for one time in the marker falling edge.

9. *Continue with the delta integration.* The delta integration, with its OUT value as result, continues to be done normally. It is done also without applying any algorithm.

In the depicted figures (6.3), (6.4), will be shown the algorithm wrote using Labview. It was wrote as SubVI, and then recalled to the main program.

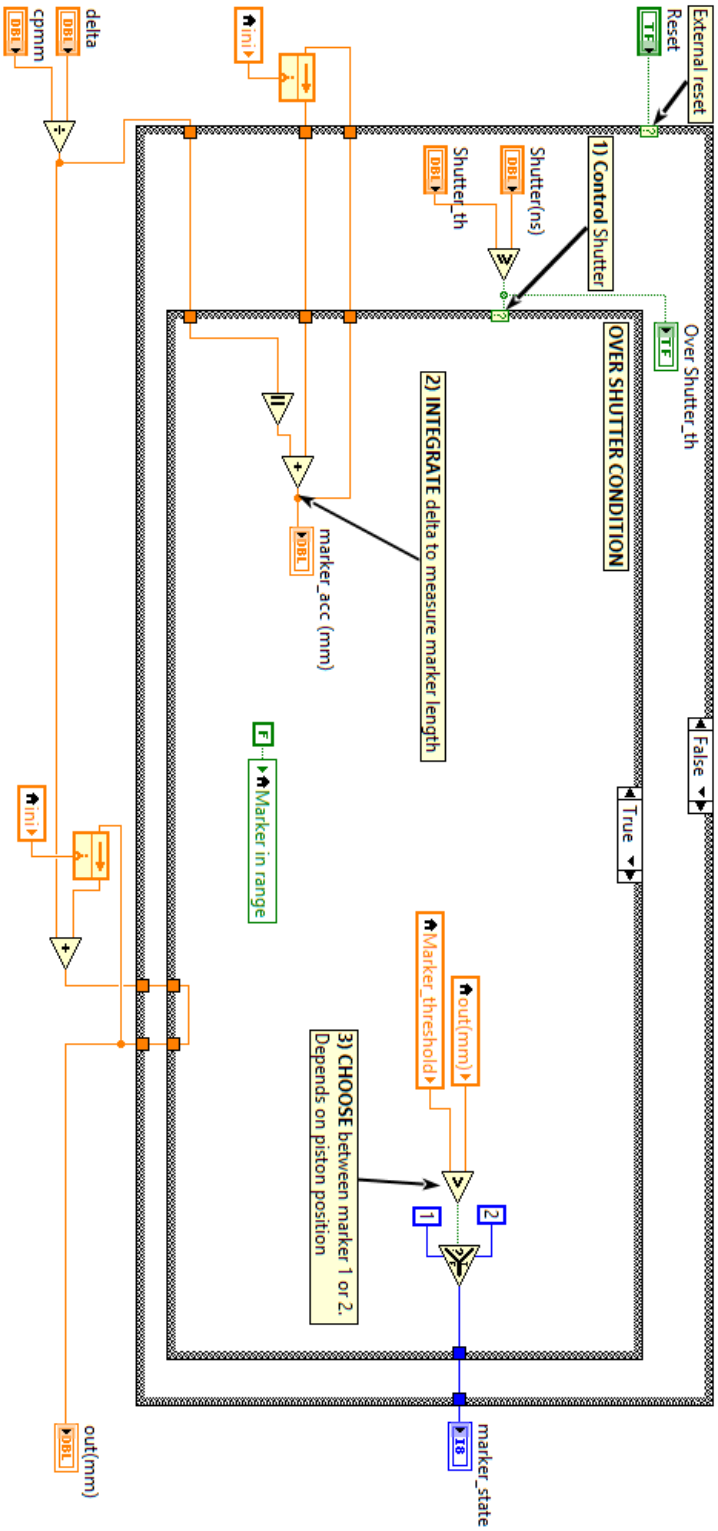


Figure 6.3: Algorithm for 2 markers: Over threshold condition

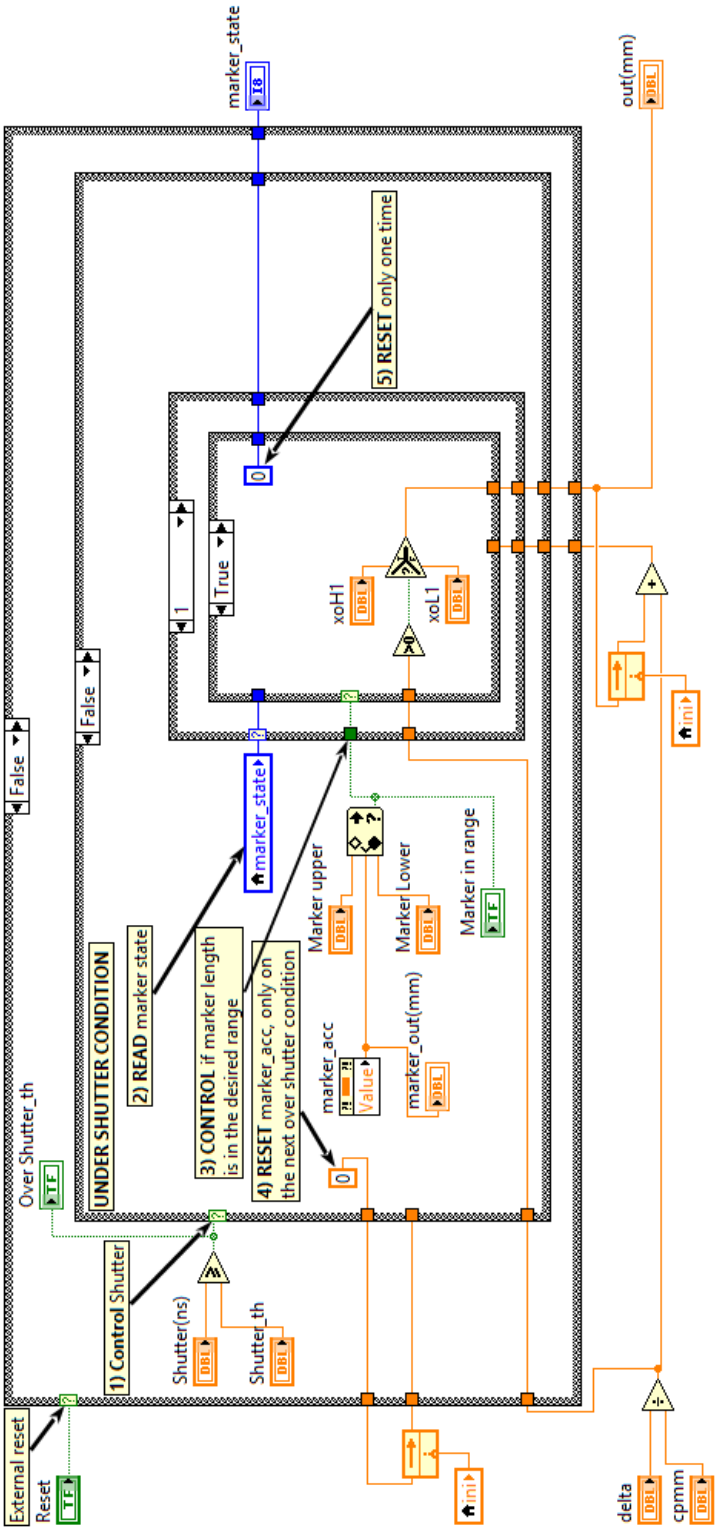


Figure 6.4: Algorithm for 2 markers: Under threshold condition

6.2 Algorithm 3

As we saw in the previous chapter, exactly in the Experiment 3, if we consider a fixed stroke the read counts provided by the sensor were different considering different tracking speeds. Precisely, the read counts were directly proportional to the speed.

So, will be proposed an algorithm that do a compensation of the read counts in three steps:

1. Do a linear fitting from a set of points, to compute the expected read of counts at different speeds
2. Use the expected read counts, to compensate the instantaneous counts read at each sampling period
3. Transform the instantaneous compensated counts in a measure in millimeters.

Doing several measurements of the same stroke, we can obtain a set of points useful to make a linear fitting of the data. In this way we can estimate the expected read counts at different velocities. The speed is obtained in terms of counts. In the depicted figure(6.5) we can see the data set and the fitting.

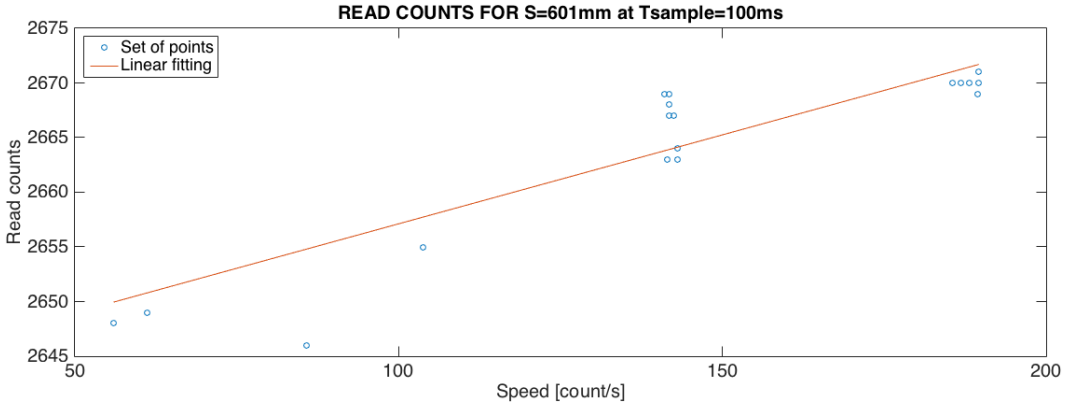


Figure 6.5: Linear fitting for read counts compensation

The fitting is linear, so the equation that represents it is:

$$y = 7.2253x + 2.64E3 \tag{6.1}$$

where y is equal to the expected read counts, x equal to the speed in terms of counts/second.

Now, we know the expected read counts for a fixed stroke. The second step is to use this information for the instantaneous counts read by the sensor at each cycle. So, to do this we can calculate the effective sensitivity thus the counts/mm that we expected. It is done dividing the read counts by the fixed stroke.

$$cpmm = \frac{y}{stroke} = \frac{ExpectedReadcounts}{stroke} \tag{6.2}$$

Then, to calculate the instantaneous compensated measure, we can simply divide the instantaneous measure, read at each cycle, with his effective cpmm. The istantaneous measure now is in millimeters.

$$Xcompensated(mm) = \frac{Xmeasure(count)}{cpmm} \tag{6.3}$$

Finally, to calculate the displacement we have to integrate the Xcompensated, that is computed at each sampling period.

In the depicted figure (6.6) will be shown the algorithm wrote using Labview. It was wrote as SubVI, and then recalled to the main program.

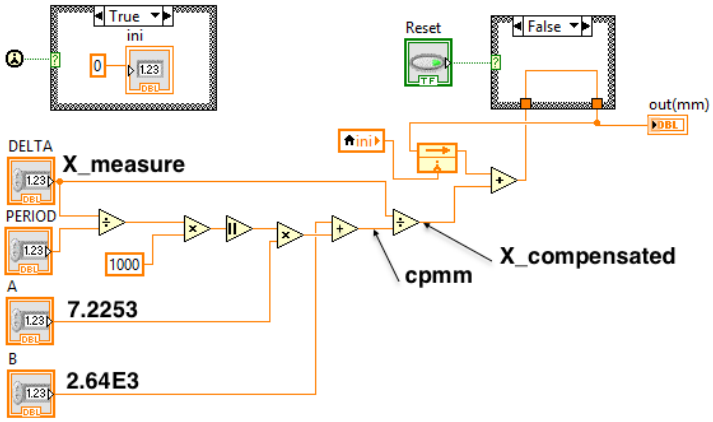


Figure 6.6: Algorithm for read counts compensation: Block diagram

6.3 Results with Algorithm 1

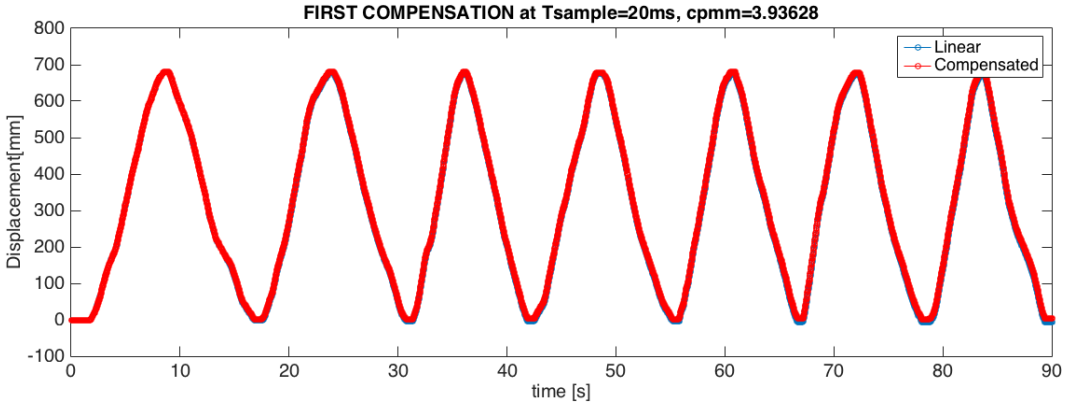


Figure 6.7: Displacement with First algorithm

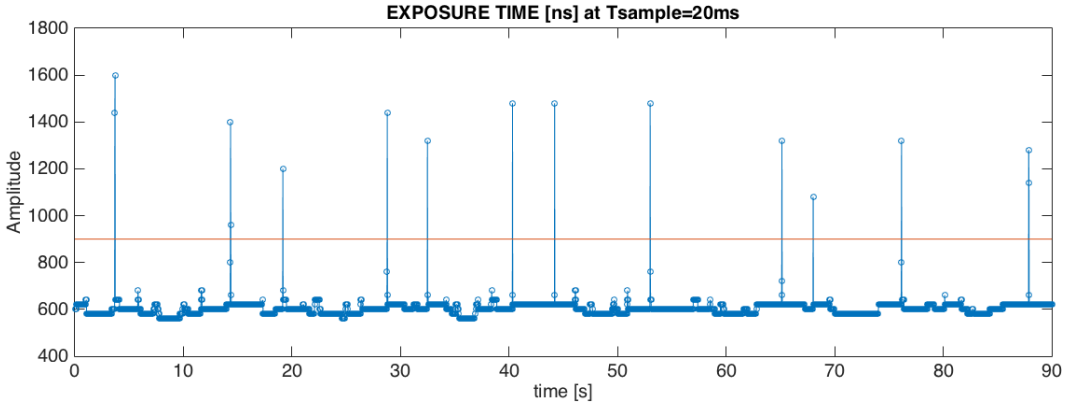


Figure 6.8: Detection of first marker at 181.5mm

As we can see in the figure (6.7), the algorithm detects only the first marker positioned at 181.5mm. However, the detection fail 2 times in 14 crosses over the first marker. It is due to the high Tsample, that worsens the detection. The results of the first algorithm confirms that the algorithm is reliable because it compensates the drifting evolution of the linear signal, however a second marker detecting can be useful to make the algorithm robust.

6.4 Results with Algorithm 2

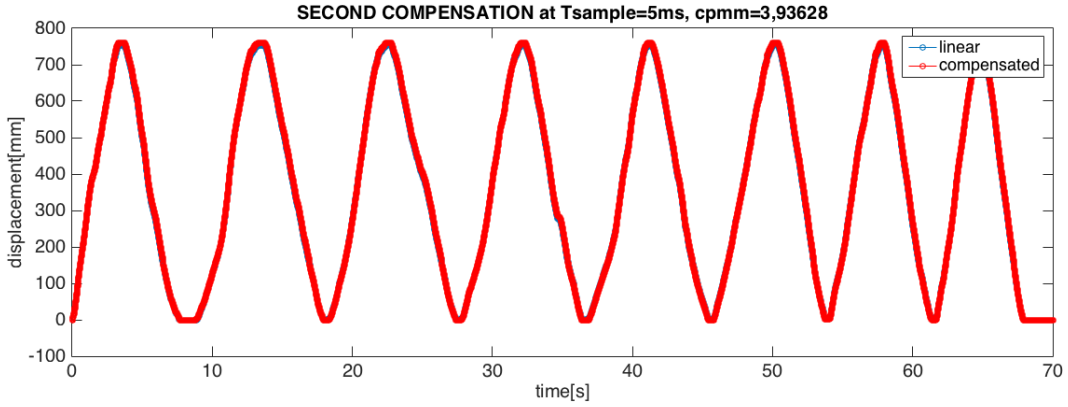


Figure 6.9: Displacement with Second algorithm

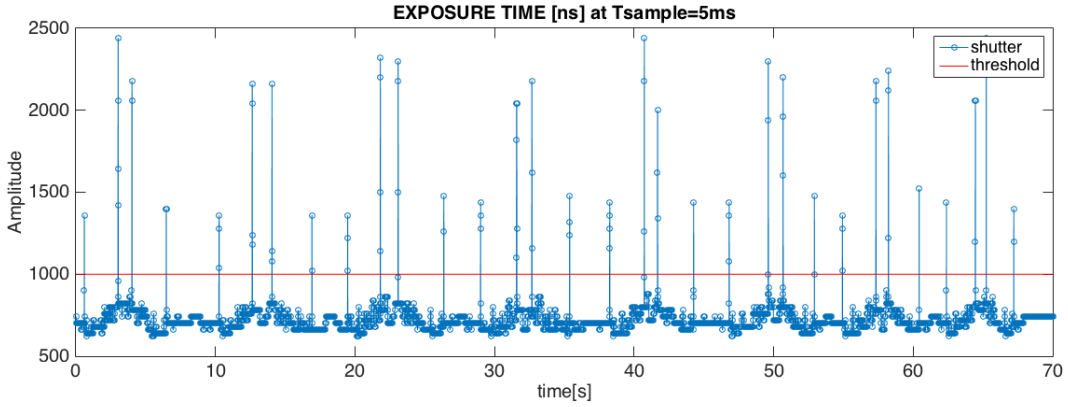


Figure 6.10: Detection of first marker at 181.5mm, and second marker at 746.5mm

As we can see in the figure (6.9), the algorithm detects the first marker positioned at 181.5mm, and the second marker positioned at 746.5mm. The detection didn't fail in any case and the Tsample was 5ms. The results of the first algorithm confirms that the algorithm is reliable and robust because it compensates the drifting evolution of the linear displacement.

6.5 Results with Algorithm 3

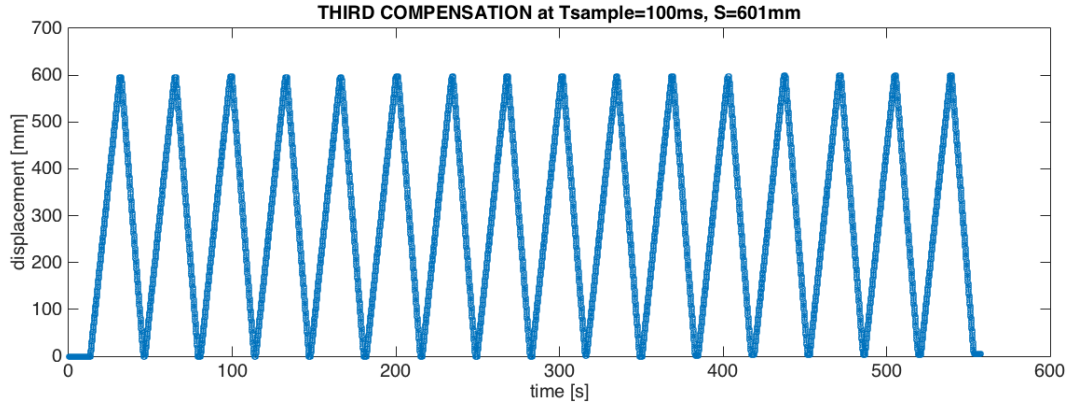


Figure 6.11: Read count compensation for 16 back and forth displacements

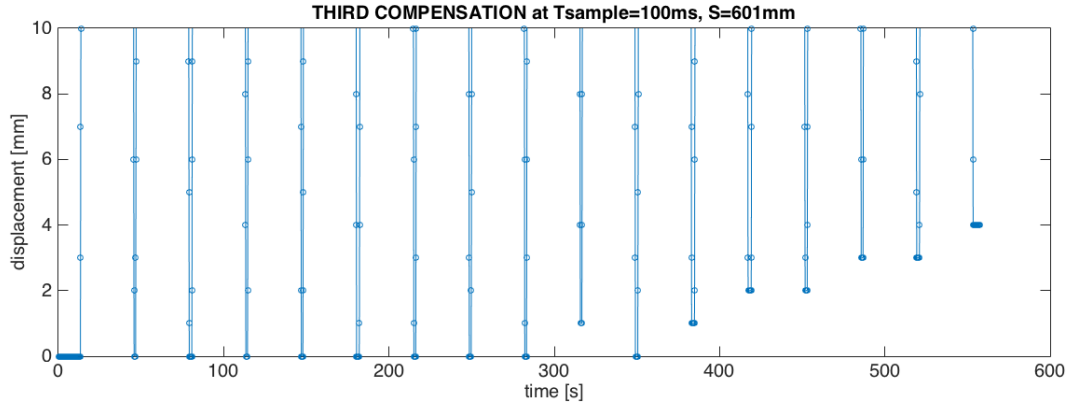


Figure 6.12: Drift evolution in time

As we can see in the figure (6.11), the algorithm compensate the instantaneous read counts acquired by the sensor at each cycle. The algorithm was tested in 16 back and forth displacements of a fixed stroke of 601mm. For the first 11 back and forth cycles the algorithm compensate the measurement very well, and forth the last 5 back and forth cycles it fails, and the measure starts to drift during the time. To enhance the algorithm it has to be acquired for sure more data sets to do a best fitting and probably it can be a good solution to develop a recursive algorithm for obtaining better results.

Chapter 7

Conclusions

In the present thesis were described the experiments done in the Optical Measurements Laboratory from Politecnico di Milano. The experiments focused in the development of a measurement system for the calculation of the linear and angular position on piston rods using optical flow sensors, precisely with optical mouse sensors. With a second priority was developed a system able to measure the circumference of big pipes. The goals that we need to reach, prefixed before the development of the systems, were to develop low cost, compact, temperature resistant, robust and reliable systems.

The development of the piston rod system initiates with the develop of the hardware in order to transmit the necessary data to the pc. Then was done an analyzing and comparison of the possible sensors that fulfill our requests. Once chosen the sensor, it was done a deep analysis to understand the sensor limits in order to propose feasible algorithms that corrects the measurement. In certain measurement conditions, the obtained results confirmed that the developed system for piston rods provides a robust and reliable measure, giving an uncertainty on the stroke measurement of about 1ppm or slightly below. This uncertainty can be obtained applying the marker algorithms proposed on the previous chapter. On the other hand, the marker algorithms needs absolute references that increase slightly the cost of the system because the absolute references were done by laser marker techniques. However the placing of laser markers depends on the application that the cylinder is proposed work. We should need to place several or few markers and it can produce a further cost to our system. In addition, it was proposed an alternative algorithm that doesn't need absolute references. This alternative algorithm aims to do a compensation of the read counts by the sensor. The algorithm was stable for a certain period of time and then failed. That result was probably due to the need of more sets of points that could enhance the data fitting, necessary

to develop the algorithm.

The develop of the pipe system initiates with the study of the optical theory. We needed to understand if changing the optical system with other lenses, could provide a reliable measure of the pipe circumference. The requirement to change the optical systems came from the necessity to did the measure far from the pipe surface, precisely at 50mm. We expected that having a rough pipe surface helps the optical mouse sensor to get good results, however with the lenses under test we didn't obtain reliable results.

For a further research we proposed:

- To use the OTS sensor with a laser illumination system, if there is no constraint of power consumption, in order to show up better the surface textures to the CMOS camera.
- Try to enhance the read counts algorithm, doing a best acquiring of data sets in order to make a best data fitting.
- Develop a system with two or more sensors, in order to reduce the average errors on measurement. It can be a feasible solution to renounce to laser markers.
- With the OTS sensor, that has a working temperature from 0 to 40 degrees, can be used a device that use the thermoelectric effect for cooling the sensor.
- For the pipe system, can be used aspheric lenses to reduce the spherical aberrations produced when the light rays finishes at different focus points.

Appendix A

Optical Sensor's Specifications

ADNS-9800
LaserStream™ Gaming Sensor

AVAGO TECHNOLOGIES

Data Sheet

Lead (Pb) Free
RoHS 6 fully
compliant

Description

The ADNS-9800 LaserStream gaming sensor comprises of sensor and VCSEL in a single chip-on-board (COB) package. ADNS-9800 provides enhanced features like programmable frame rate, programmable resolution, configurable sleep and wake up time to suit various PC gamers' preferences.

The advanced class of VCSEL was engineered by Avago Technologies to provide a laser diode with a single longitudinal and a single transverse mode.

This LaserStream gaming sensor is in 16-pin integrated chip-on-board (COB) package. It is designed to be used with ADNS-6190-002 small form factor (SFF) gaming laser lens to achieve the optimum performance featured in this document. These parts provide a complete and compact navigation system without moving part and laser calibration process is NOT required in the complete mouse form, thus facilitating high volume assembly.

Theory of Operation

The sensor is based on LaserStream technology, which measures changes in position by optically acquiring sequential surface images (frames) and mathematically determining the direction and magnitude of movement. It contains an Image Acquisition System (IAS), a Digital Signal Processor (DSP), and a four wire serial port. The IAS acquires microscopic surface images via the lens and illumination system. These images are processed by the DSP to determine the direction and distance of motion. The DSP calculates the Δx and Δy relative displacement values. An external microcontroller reads the Δx and Δy information from the sensor serial port. The microcontroller then translates the data into PS2, USB, or RF signals before sending them to the host PC or game console.

Features

- Small form factor chip-on-board package
- Dual power supply selections, 3 V or 5 V
- VDDIO range: 1.65 – 3.3 V
- 16-bits motion data registers
- High speed motion detection up to 150 ips and acceleration up to 30 g
- Advanced technology 832-865 nm wavelength VCSEL
- Single mode lasing
- No laser power calibration needed
- Compliance to IEC/EN 60825-1 Eye Safety
  - Class 1 laser power output level
  - On-chip laser fault detect circuitry
- Self-adjusting frame rate for optimum performance
- Motion detect pin output
- Internal oscillator – no external clock input needed
- Enhanced Programmability
  - Frame rate up to 12,000 fps
  - 1 to 5 mm lift detection
  - Resolution up to 8200 cpi with ~200 cpi step
  - X and Y axes independent resolution setting
  - Register enabled Rest Modes
  - Sleep and wake up times

Applications

- Corded and cordless gaming laser mice
- Optical trackballs
- Motion input devices

CAUTION: *It is advised that normal static precautions be taken in handling and assembly of this component to prevent damage and/or degradation which may be induced by ESD.*

PixArt Imaging Inc.

ADBS-A350

Optical Finger Navigation

Data Sheet

Lead (Pb) Free
RoHS 6 fully compliant

Description

The ADBS-A350 sensor is a small form factor (SFF) LED illuminated optical finger navigation system. It is a low-power optical finger navigation sensor with low-power architecture and automatic power management modes, making it ideal for battery-and power-sensitive applications such as mobile phones.

The ADBS-A350 is capable of high-speed motion detection – up to 20 ips. In addition, it has an on-chip oscillator and integrated LED to minimize external components.

There are no moving parts, thus provide high reliability and less maintenance for the end user. In addition, precision optical alignment is not required, facilitating high volume assembly.

The sensor is programmed via registers through either a serial peripheral interface or a two wire interface port. It is packaged in a 28 I/O surface mountable package.

The ADBS-A350 is designed for use with ADBL-A321 lens.

The ADBL-A321 lens is the optical component necessary for proper operation of the sensor.

Theory of Operation

The ADBS-A350 is based on Optical Finger Navigation (OFN) Technology, which measures changes in position by optically acquiring sequential surface images (frames) and mathematically determining the direction and magnitude of movement.

The ADBS-A350 contains an Image Acquisition System (IAS), a Digital Signal Processor (DSP), and a communication system.

The IAS acquires microscopic surface images via the lens and illumination system. These images are processed by the DSP to determine the direction and distance of motion. The DSP calculates the Δx and Δy relative displacement values.

The host reads the Δx and Δy information from the sensor serial port if a motion interrupt is published. The microcontroller then translates the data into cursor navigation, rocker switch, scrolling or other system dependent navigation data.

Features

- Low power architecture
- Surface mount technology (SMT) device
- Self-adjusting power-saving modes for longer battery life
- High speed motion detection up to 20 ips
- Self-adjusting frame rate for optimum performance
- Motion detect interrupt
- Finger detect interrupt
- Soft click and Tap detect interrupt
- Single Interrupt pin
- Optional PWM output for LED illumination
- Optional switch input for center click function
- Internal oscillator – no clock input needed
- Selectable 125, 250, 500, 750, 1000 and 1250 cpi resolution
- Single 1.8 V supply voltage for analog and digital
- Internal power up reset (POR)
- Selectable Input/Output voltage at 1.8 V or 2.8 V nominal
- 4-wire Serial peripheral interface (SPI) or Two-wire interface (TWI)
- Integrated chip-on-board LED with wavelength of 870 nm

Applications

- Finger input devices
- Mobile devices
- Integrated input devices
- Battery-powered input device

PixArt customers purchasing the ADBS-350 OFN product are eligible to receive a royalty free license to our US patents 6977645, 6621483, 6950094, 6172354 and 7289649, for use in their end products.

***CAUTION:** It is advised that normal static precautions be taken in handling and assembly of this component to prevent damage and/or degradation which may be induced by ESD.*

PixArt Imaging Inc.
E-mail: fae_service@pixart.com.tw

PixArt Imaging Inc.

Optical Track Sensor (OTS)

Preliminary Data Sheet

Lead (Pb) Free
RoHS 6 fully
compliant

General Description

The is Optical Track Sensor (OTS) using optical navigation technology for motion reporting purposes. The OTS integrates LED source and optical sensor in a single small form factor package. The OTS sensor has built-in image recognition engine, which does not require additional hardware or any special markings on surface. The OTS sensor offers a direct SPI output together with motion interrupt signal, provides easy integration with host system.

Features

- Integrated 16 pin molded lead-frame DIP package
- Four wire serial port interface (SPI)
- High speed motion detection 150ips (typical) and acceleration 30g(max)
- High resolutions up to 8200cpi with 100cpi step size
- External interrupt output for motion detection
- Internal oscillator — no clock input needed
- Built in Rest modes for power saving

Applications

- provides highly accurate motion detection in both linear and cursor modes.

Key Specification

Power Supply Range	1.8V – 2.1V
Optical Lens	1:1
Interface	4 wire serial port interface
Frame rate	Up to 8200 frame per second
Speed	150ips (typical)
Resolution	8200cpi
Package	16 pin molded lead-frame DIP package

Appendix B

Keyence Specifications

■ LK-G157/LK-G152

Modello		LK-G157/LK-G152	
Modo regolazione		Riflessione diffusa	Riflessione speculare
Distanza di riferimento		150 mm	147,5 mm
Campo di misura *1		±40 mm	±39 mm
Sorgente luminosa		Laser semiconduttore ad infrarossi	
	Lunghezza d'onda	655 nm (luce visibile)	
	Laser Classe	Classe 2 (CEI EN 60825-1), Class II (FDA CDRH Part 1040.10)	
	Emissione	0,95 mW	
Diametro spot (alla distanza di riferimento)		Circa 120 x 1700 µm (G157) Circa ø120 µm (G152)	
Linearità*2		±0,05% di F.S. (F.S. = ±40 mm)	
Ripetibilità *3		0,5 µm	
Frequenza di campionatura		20/50/100/200/500/1000 µs (Selezionabile tra 6 tipi)	
Indicatori LED		Vicino al centro del campo di misura: Luce verde Compreso nel campo di misura: Luce arancione Fuoricampo misura: Lampeggia in arancione	
Caratteristiche temperatura		0,01% di F.S./°C (F.S. = ±40 mm)	
Resistenza alle condizioni ambientali	Grado di protezione	IP67 (IEC60529)	
	Luce ambientale di esercizio	Lampada incandescente o fluorescente: 5000 lux max	
	Temperatura ambiente di esercizio	da 0 a +50 °C	
	Umidità ambientale di esercizio	da 35 a 85% UR (Senza condensa)	
	Resistenza alle vibrazioni	da 10 a 55 Hz, doppia ampiezza 1,5 mm, 2 ore per ciascuno degli assi XYZ	
Materiale		Alluminio pressofuso	
Peso (cavo compreso)		Circa 290 g	

*1 Valore quando si misura l'oggetto standard KEYENCE (ceramica). Quando la frequenza di campionatura è 20 µs, il valore varia da –22 (lato vicino NEAR) a –40 mm (lato vicino NEAR) nel momento di riflessione diffusa, e da +22 (lato vicino NEAR) a –39 mm (lato vicino NEAR) nel momento di riflessione speculare.

*2 Valore quando si misura l'oggetto standard KEYENCE (ceramica) nel modo standard

*3 Valore quando si misura l'oggetto standard KEYENCE (SUS) con una media di volte di 4096 alla distanza di riferimento.

Bibliography

[1] Anandan P., *Measuring Visual Motion from Image Sequences*, PhD dissertation, Univ. of Massachusetts, Amherest, MA, 1987.

[2] Beauchemin S.S, Barron J.L, *The computation of optical flow*, ACM New York, USA, 1995

[3] Barron, J. L.; Fleet, D. J. and Beauchemin, S. S., *Performance of optical flow techniques*, International Journal of Computer Vision, Volume 12, Issue 1, pp. 43-77, 1994.

[4] Bonarini , A.; Matteucci, M., Restelli, M., *A Kinematic-independent Dead-reckoning Sensor for Indoor Mobile Robotics*, Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 3750-3755, 0-7803-8463-61, Sendai, Japan, September 28 - October 2, 2004.

[5] Chhaniyara, S.; Bunnun,P.; Seneviratne, L. D., Althoefer, K. (2008.) *Optical Flow Algorithm for Velocity Estimation of Ground Vehicles: A Feasibility Study*, International Journal on Smart Sensing and Intelligent Systems, Vol. 1, No. 1, (March 2008.) pp.246-268

[6] Control Products Inc, *SL series sensors*, [Online]. Available: www.cpi-nj.com

[7] Correvit(R)-SL (2001) Non-Contact Optical Sensor for slip free measurement of longitudinal and transversal dynamics, Corrsys-Datron Sensorsysteme GmbH, 2001. [Online]. Available: www.corrsys-datron.com

[8] Fleet, D. J., Jepson, A. D., *Computation of component image velocity from local phase information*, International Journal of Computer Vision, Volume 5, Number 1, pp.77-104, 1990.

[9] Heeger, D. J., *Optical flow using spatiotemporal filters*, International Journal of Computer Vision, Volume 1, Number 4, pp. 279-302, 1988.

[10] Horn B.K.P, and Schunck, B.G., *Determining optical flow.*, Artificial Intelligence, vol 17, pp 185-203, 1981.

[11] Horn, J.; Bachmann, A., Dang, T. (2006) *A Fusion Approach for Image-Based Measurement of Speed Over Ground*, Proceedings of International Conference on Multisensor Fusion and Integration for Intelligent Systems, pp. 261-266, September 3-6, 2006, Heidelberg, Germany

[12] Marr, D , *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*, Freeman, San Francisco, 1982.

[13] MTS sensors, *Temposonics® R series position sensors*, [Online]. Available: www.mtssensors.de

[14] Ng, T.W. (2003),*The optical mouse as a two-dimensional displacement sensor.*,Sensors and Actuators, A 107, (2003.) pp.21-25

[15] Ng, T.W., Ang, K.T. (2004) *The optical mouse for vibratory motion sensing.*,Sensors and Actuators, A 116, (2004.) pp.205-208

[16] Palacin, J.; Valgan?on, I., Pernia, R., *The optical mouse for indoor mobile robot odometry measurement*, Sensors and Actuators A 126 (2006.) pp.141?147

[17] Philip H.S. Torr and Andrew Zisserman, *Feature Based Methods for Structure and Motion Estimation*, ICCV Workshop on Vision Algorithms, pages 278-294, 1999.

[18] Rota Engineer Limited, *ELA model position sensor*, [Online]. Available: www.rota-eng.com

[19] Singh A., *An estimation theoretic framework for image flow computation*, Proc IEEE ICCV,Osaka, pp. 168-177, 1990.

[20] Sorensen, D. K., *On-line Optical Flow Feedback for Mobile Robot Localization/Navigation*, MSc Thesis, 2003, AM University, Texas, USA

[21] Takcs, T.; Klmn, V. (2007),*Optical Navigation Sensor, Incorporating vehicle dynamics information in mapmaking*, Proceedings of ICINCO 2007, pp. 271-274, Angers France, 2007

[22] OSMES (2004) OSMES the optical speed measurement system Siemens Transportation Systems 2004. [Online]. Available: www.transportation.siemens.com