Symbol	Description
t_0	Initial Time
$v(t_0)$	Initial Velocity

To get the position signal from velocity, is used the followed formula:

$$x(t) = x(t_0) + \int_{t_0}^{t} v(\tau)\,\mathrm{d}\tau. \tag{2}$$

Where:

Symbol	Description
t_0	Initial Time
$x(t_0)$	Initial Position

Therefore, for a double integration of the acceleration, the two initial conditions (velocity and position) must be known to avoid integration errors. However, the only way to get these initial conditions is through direct measurement, which is often impractical or unobtainable. By data filtering, it is developed an approach that does not require knowledge of initial conditions. To perform a numerical integration it is necessary to choose one integration algorithm of the many existing. The acceleration signal is sampled, making it a discrete function of time having a sampling frequency, f_s, associated with it. The easiest way to perform numerical integration is to use the rectangular integration method. This method uses an accumulator to sum all past sampled inputs and the current input sample and divide by the sampling rate. Rectangular integration is represented by the following difference equation:

$$y(n) = 1f_s \sum_{k=0}^{n} x(n-k) = y(n-1) + 1f_s x(n)$$

Another numerical integration method uses the trapezoidal rule. The results are more precise with this method than the rectangular method. The difference equation for trapezoidal integration is:

$$y(n) = y(n-1) + 12f_s[x(n-1) + x(n)], n > 0$$

The figure below show the difference using both methods.

[scale=0.55]RectangularInte [scale=0.55]TrapezoidalIntegration

Figure 1: Integration using Rectangular (left) and Trapezoidal (right) methods of Sine Wave.

The integration was carried out using the trapezoidal method by the MATLAB suite. But without adequate cleaning of the signal, the result presents very error, caused by the drift and noise.
Various signal cleanup procedures will be discussed. The figure below shows the result of a raw signal integration using trapezoidal method.

[Acceleration][scale=0.11]RawAccelerometer
[Velocity][scale=0.11]VelocityIntegration
[Displacement][scale=0.11]DisplacementIntegration

Figure 2: The result of double integration of a Raw Signal

As is possible see, the result is not realistic, considering it is vertical acceleration, so it is necessary to perform various steps to fix and clean the signal. First of all, the signal reorientation will be carried out respect the axes of the vehicle, and subsequently, various filter operations to clean the signal.

1 Accelerometer Reorientation

The Cartesian reference system of the phone must be aligned with the vehicle reference system, to detect the vehicle motion correctly. As shown in the figure below.

[scale=0.7]phonecarorientation

Figure 3: Correct alignment of smartphone respect to the vehicle cartesian frame.

Smartphone accelerometer detects the following accelerations: a_{x_p} , a_{y_p} and a_{z_p} . To determine the accelerations felt by the vehicle and locate road surface anomalies. The accelerometer must detect what happens in the direction perpendicular (Z axes) to the vehicle [**?**]. Respectively the X_p axis identifies the longitudinal direction, Y_p axis the transverse direction and Z_p the perpendicular direction respect to the xy plane.
To detect road anomalies, the direction of the z-axis must correspond to the direction of the z axis. If this condition subsists, the accelerometer is well oriented, contrarily it is not well oriented and needs to be reoriented. But even if starting from a precise orientation condition, during travel the phone may be moving, or due to unexpected vehicle movements, travel of climbing, downhill, curve, all of these causes could affect the misalignment of the smartphone frame respect to the vehicle frame.
The reorientation can be performed by the Euler Angles. Three angles that allow to defining the orientation in space of any body through a succession of elementary rotations.[**?**] The XYZ sequence was defined,a rotation around the x axis by an angle α (roll angle), one around the y axis by β (pitch angle) and one around the z axis by γ (yaw angle).
The equations that allows to reoriented data by the α, β, γ, angle are:[**?**]

$$a_{x_{reor}} = \cos(\beta)a_{x_p} + \sin(\beta)\sin(\alpha)a_{y_p} + \cos(\alpha)\sin(\beta)a_{z_p} \tag{3}$$

$$a_{y_{reor}} = \cos(\alpha)a_{y_p} - \sin(\alpha)a_{z_p} \tag{4}$$

$$a_{z_{reor}} = -\sin(\beta)a_{x_p} + \cos(\beta)\sin(\alpha)a_{y_p} + \cos(\beta)\cos(\alpha)a_{z_p} \tag{5}$$

The figure below, show an example of reoriented data.

[Raw][scale=0.11]raw
[Reoriented][scale=0.11]reoriented

Figure 4: The result of reorientation of raw data

2 Data Filtering

As can be seen in the figure 4.2, a data filtering operation is needed to clear the signal and perform better integration. It is necessary to apply:

- A series of preliminary filters for removing unimportant information.
- A digital filter for removing certain frequency components that cause the error at the stage of integration.

2.1 Preliminary Filtering

During signal capture, it is important to consider some information that can be removed, since they are not relevant to the final calculation.
These principally regard:

- The background noise generated by engine vibrations.
- Acceleration components recorded at the moment the vehicle is stationary, so when its velocity is equal to 0 .

Or:

$$a_t = \sqrt{1k(a_{t_i}^2 + a_{t_{i+1}}^2 + ... + a_{t_{i+k}}^2)}$$

$$if \quad a_{min_{th}} <= a_t <= a_{max_{th}}$$

Where k is the number of a predefined window, and all the values that falls into the window indexes will be considered.
So the final value a_t will be calculated as the Root Mean Square of that values.

The figure below shows an example of applying the filter to a series of data subject only to the reorientation operation4.1.

[Acceleration][scale=0.1]WithRumors [Without Engine Vibrations][scale=0.1]NoRumors [Overlapping][scale=0.1]EngineRumors
The original signal (a) is represent in blue, the filtered signal is represent in orange (b).

Figure 5: Application of Engine Vibrations Filter

As is possible to see the signal when filtered it is much lighter, the background noise component produced by the engine is very resonant during the acquisition process, and with this process it is possible to delete it.

2.1.2 Zero Velocity Filter

This filter brings attention to the cancellation of certain acceleration values that are associated with zero speed values. Due to several factors, it may happen that while the vehicle is stationary, because the registration starts even before the vehicle is in motion, or we are in intense traffic situations, stopped at a traffic light, and all other situations that may occur, making us stopped while the smartphone continues to read the data. If any of these conditions occur, the acceleration values can be cancelled, because they would not help us to understand the condition of the road surface.
It is possible understand that speed at a given time (t) is equal to 0 , thanks to GPS, beacuse during smartphone recordings, gives us the speed of travel. Considering both signals at generic time (t), acceleration (a_t), and velocity (v_t), the data series is so modified:

$$\begin{cases} a_t = 0; & if \quad v_t = 0 \\ a_t = a_t; & if \quad v_t \neq 0 \end{cases}$$

The figure below shows an example of applying this filter to a series of data subject before to Engine Vibrations Filter4.2.1.1.

[Acceleration][scale=0.099]NoRumors [No Zero Velocity][scale=0.074]NoVelocity [Overlapping][scale=0.1]NoVelocitySovrapposte
The original signal (a) is represent in blue, the filtered signal is represent in orange (b).

Figure 6: Application of Zero Velocity Filter

2.2 Digital Filtering

For IRI calculation, vertical displacement is needed, so the accelerometer data must be subject to a double integration process.

Unfortunately, accelerometers have an undesired phenomenon named drift associated with them produced by a small DC component[3] in the acceleration signal.

Ideally, there should be no DC from the accelerometer for the measurement of a vibration. The presence of drift can direct to high integration errors. If the acceleration signal was integrated without any proper filtering, the output could become unlimited over time. The figure 4.2 shows what usually occurs to an acceleration signal after a double integration. Figure 4.2, is an example of acceleration signal that has a negative DC.

To resolve the drift problem, a filter can be used to remove the DC component from the acceleration signal. Through filtering before integration, drift errors can be eliminated.

For the initial conditions as discusses in 4, a solution is to use filtering.

After the acceleration signal is integrated, it will possibly have a DC component. Again a filter can be used to remove that DC component of the signal. Furthermore, after the velocity signal is integrated to get position, the position signal can be filtered as well.

Filtering is a particular frequency process that attenuates certain bands of frequencies while passing others. These filters will pass the high-frequency content of a signal while rejecting the low. The specifications of a filter are its cutoff frequency, pass-band attenuation, and stop-band attenuation.

It is convenient if the filters are identical to each other to simplify the design.

There are two types of filters in the digital area: Infinite Impulse Response (IIR) filters and Finite Impulse Response (FIR) filters.

This is applicable as long as the filter does not attenuate frequencies in the signal band.

2.2.1 Finite Impulse Response

Finite Impulse Response, (FIR), is a type of digital filter characterised by a finite response, that, it is cancelled at a finite time.

A FIR filter is describedd by the following difference equation:

$$y[n] = b_o x[n] + b_1 x[n-1] + ... + b_N x[n-N] = \sum i = 0^N b_i x[n-i]$$

Where: $x[n]$ is the input signal, $y[n]$ is the output signal, N is the filter order, and b_i is the value of the impulse response at instant i.

This filter is useful for the double integration process. It is recommended to use it because its phase response is linear, which is desired because different frequencies passing through the filter will have the same time delay.

A disadvantage is that the order can be very high, and lead to excessive computations.

For application to a vehicle road test, there is an interest in processing low frequency signals. So the filter must have a low cutoff frequency with a clear transition band, making the order of the filter high.

Moving Average

An example of FIR filter is the moving average, commonly used in road pavement profiles, that defines the point A_i as the average of the points close to that one, for a base window of length n [**?**], defined like follow:

$$A_i = 1n \sum_{k=i}^{k=i+n} A_k$$

An example of application of the moving average filter is shown in the figure below:

[Unfiltered][scale=0.1]NoMovingFiltered [Filtered][scale=0.1]MovingFiltered [Overlapping][scale=0.15]TotalFiltered
The original signal (a) is represent in blue, the filtered signal is represent in orange (b).

Figure 7: Application of Moving Average Filter

2.2.2 Infinite Impulse Response

In signals theory, Infinite Impulse Response (IIR) is a dynamic system whose impulsive response is not nothing for an infinite amount of time.
IIR filtering, is an alternative approach, that uses a recursive difference equation to represent the filter.

$$y[n] = a_0(\sum_{i=0}^{P} b_i x[n-1] + \sum_{j=1}^{Q} a_j y[n-j])$$

The output is written as a combination of present and past inputs and past outputs.

3 GPS points division

Another problem to fix is the way geolocation points are identified.
As discussed in Chapter 3, (GPS section, page), GPS records the data at a frequency of 1 Hz (every second), however, the acceleration data as we have seen in our case are recorded at 100 Hz (every 10 ms).
For every GPS recording, 100 acceleration data will be registered. This puts us in front of a problem because if we want to map the road surface correctly, GPS data need to be very close to each other.
To understand better this factor, two different GPS coordinates were identified:

$$Start = (Latitude_{start}, Longitude_{start})$$
$$End = (Longitude_{end}, Longitude_{end})$$

Show on the figure below:
FIGURA DA AGGIUNGERE
As we can see, we have two different GPS points, but we have a data accumulation at the *Start* point, in our case equal to 100 measurements and each of these has *Start* point as GPS coordinate, the situation is analogue to the *End* point.
Start point recordings refer to all $Start - End$ segment, but the intermediate coordinates of the segment are not recorded, and that is what is needed to properly locate the road surface conditions.
Therefore, a methodology has been developed, which from two different GPS points, allows determining the intermediate points in the segment. As shown in the code below.
Redefine Latitude And LongitudeLatitude[],Longitude[] Create two collections of new Latitude and Longitude
NewLatitudeCollection[]
NewLongitudeCollection[]
index = 1
i = 1
y2 number of Latitude elements
FirstGPSCoordinate = Latitude[i], Longitude[i]
SecondGPSCoordinate = Latitude[y], Longitude[y]
distance FirstGPSCoordinate,SecondGPSCoordinate $\neq 0$
distance function, use haversine formula, that is next explained
NewLatitudeCollection [index] Latitude[i]
index++

Data Analysis As discussed in Chapter3 in the Error section(3.4 on page 33). IMU, during measurement suffer from accumulated error.33The measurement along the pathway of road surface consists of a sampling of measurements at specific time intervals (usually in the order of milliseconds).So the road profile can be seen as an electronically acquired digital signal and consequently it is necessary clean up the signal from the noise[1], it may also be useful to delete certain information in the signal that is not of interest, or correct it from the drift (3.4).This cleaning procedure, also called filtering,it is used in the theory of signal analysis. It is an essential aspect in the evaluation of the profile, [?]. In brief, a digital filter represents a procedure that transforms a series of numbers into a new series of numbers, [?].From the measurement obtained by the sensors, particular attention is given to the vertical acceleration signal and to the rotation vectors.The rotation vectors will be useful for reorienting the signals obtained from the smartphone[?] respect to the vehicle axes, while the vertical acceleration signal will be subject to various filtering operations depending on the final index to be obtained.First of all, it is necessary to pay some attention to some dynamics[2] physical principles , about the operations that can be done, on the acceleration signal to get the position and vice-versa.Given a position versus time of an object, $x(t)$, the velocity, $v(t)$, can be found by taking the first derivative. $v(t) = dxdt$ Acceleration, $a(t)$, can be found by taking the second derivative of position or first derivative of velocity. $a(t) = d^2xdt^2 = dvdt$ However, is interesting to reverse this process and find the position signal given an acceleration signal. To do that, a double integration must be performed on the acceleration signal. In principle, using double integration on an acceleration signal to get a position signal, the initial position and initial velocity must be known. After the first integration, the initial velocity should be added to the result, as the initial position should be added after the second integration. These operations are illustrated in the following equations $v(t) = v(t_0) + \int_{t_0}^{t} a(\tau)\, d\tau$ (1)Where: