Foucault Pendulum Tracking System Using a Raspberry Pi 4

Raffaelli Claudia, Francesco Scandiffio

claudia.raffaelli@stud.unifi.it, francesco.scandiffio@stud.unifi.it

Department of Information Engineering
University of Florence
Via di Santa Marta 3, Florence, Italy

Overview

Introduction - Aim of the project

Main goal

The aim of this project is to develop a program capable of performing the tracking of a **Foucault pendulum**, with the purpose of analysing its motion.

We will see:

- The basic requirement of this project.
- The main approaches implemented searching for the best solution to the problem.
- The practices put in place while improving the system.
- What is the output produced.
- The graphical interface and operation that can be performed.

Introduction - System Requirements

The system needs to comply with the following **requirements**:

- It has to be a **tracking system** capable of detecting in real-time the centre of the pendulum, providing its (x, y) coordinates.
- It must be **accurate** in the extraction of the coordinates.
- It has to be **fast** enough to collect in **real-time** an adequate number of points for each oscillation.
- Has to **store** in a file the collected positions as a function of **time**.
- The saved coordinates need to take into account the **perspective correction**.
- Use the points to draw a **2D graph** of the pendulum trajectory.

Introduction - Tracking System Overview

The general Foucault pendulum:

- If observed for a short time is identical to a simple pendulum.
- Its oscillation trajectory rotates on a plane due to the terrestrial rotation.
- It moves at a predictable rate depending on the latitude.

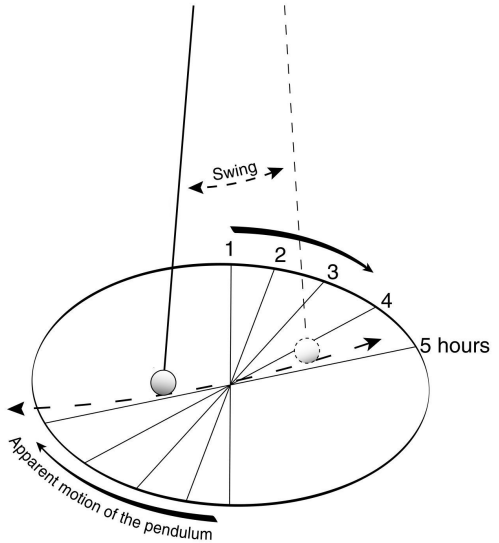


Figure 1: Foucault pendulum.

The **hardware** consists of:

- A **Foucault Pendulum** placed on an electromagnetic coil.
- A **Raspberry Pi** 4.
- A **camera** placed over the pendulum.
  - Since the device is slightly **toed-in**, this will require the application of a perspective correction on the frames.
  - Also the camera is of type **Rolling Shutter** [1]: the frame pixels are not acquired at the same time and this can lead to some distortions.

Tracking Algorithms: Chromatic Thresholding

Thresholding

An image processing technique which provides a binary segmentation of the Input image, i.e. it generates a black and white image in which the white pixels are all related to the content of interest to be extracted.

- A **red marker** has been placed on the pendulum centre. The idea is to filter out from the frame all that is not red, thus locating the centre.
- The frames are converted from BGR model to **HSV** colour space which is more robust against noise [2].

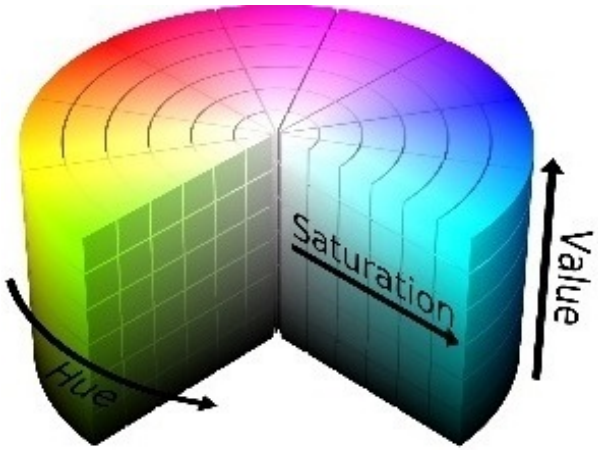
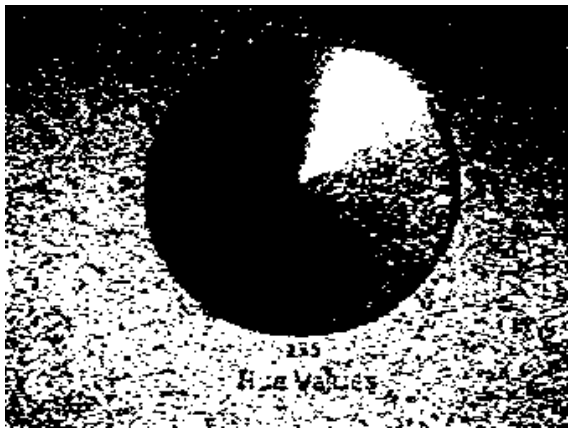


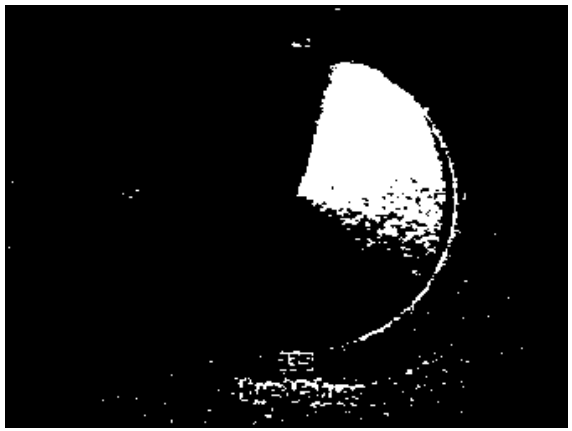
Figure 2: HSV Colour Space.

HSV Color Space

- Separates the information about the colour Hue from the information about its intensity and saturation.
- Theoretically offers great robustness against noise, such as shadows and light reflections.
- Actually, the light conditions affect the perceived colour tone.



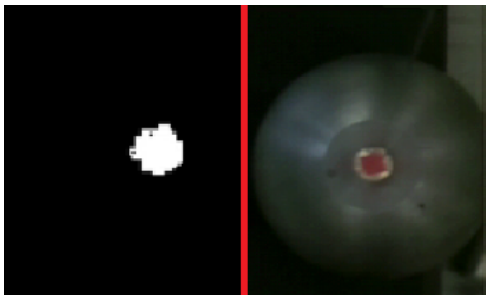
(a) Thresholding applied in low-light condition.



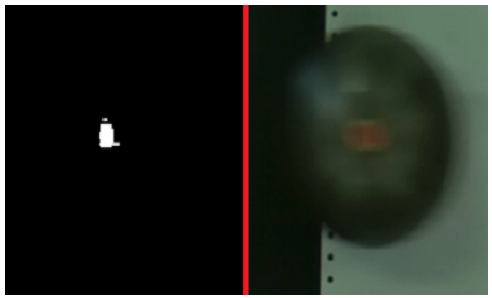
(b) Thresholding applied in well-lit condition.

Tracking Algorithms: Thresholding Results

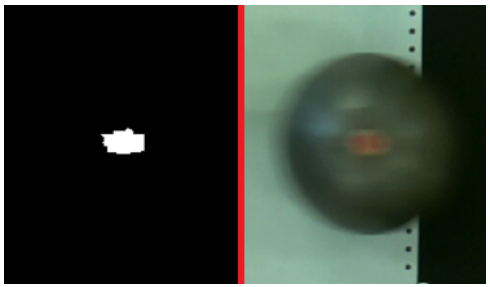
- Considerably sensitive to ambient brightness:
  - **False positives** due to the angle of incidence of light.
  - **True negatives** due to shadows.
- Ineffective when figures are deformed:
  - **Rolling Shutter** technology alters the shapes of fast-moving objects with consequent loss of precision.



(a) Ground Truth



(b) Thresholding applied to the pendulum while covered by shadows.



(c) Thresholding applied to the pendulum in motion and in a well-lit condition.

Tracking Algorithms: Circle Detection

Circle Detection

The process of locating circles in images. This task is carried on using the feature detection technique **Circle Hough Transform** (CHT) [3].

Preparatory steps:

- The frame is first converted to a grey-scale image.
- A **Gaussian blur** is applied to the figure, in order to remove noise and smooth out unnecessary details i.e. it removes irrelevant **edges** (points for which there is a sharp change of colour).

Now is applied the CHT of OpenCV that:

- For each remaining edge point (x, y) found, defines a circle with that centre, and fixed radius.
- The intersection point of all such circles is indicative to the centre point of the real circle.

Tracking Algorithms: Circle Detection - cont.

How this method performs:

- From a theoretical point of view, this method seems to work well, but besides the pre-processing to the image, the original frame has to be of **good quality** in order for it to behave properly.
- Again, the camera is of type **Rolling Shutter**, and the frames appear deformed. As a consequence, even the figures inside the image are distorted.

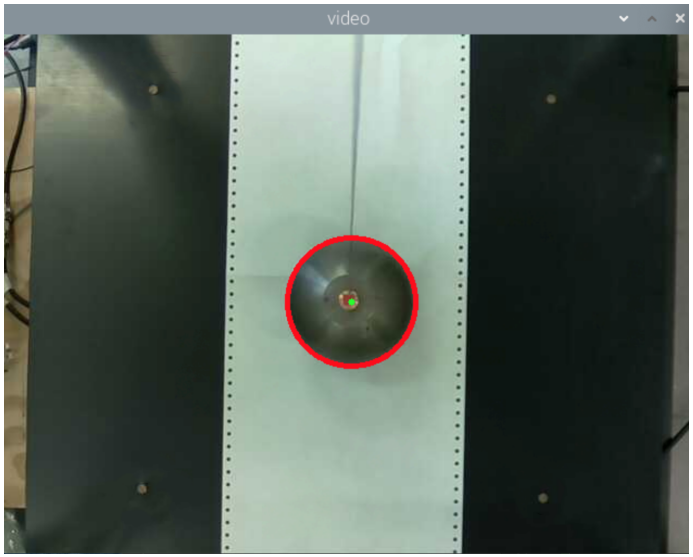


Figure 5: Circle detection applied to still pendulum frame.

- It performs well with still frames and non fast moving objects, but this is not our case.

Tracking Algorithms: Template Matching

Template Matching

An image processing technique in which the recognition is achieved by searching within an Input image for the group of pixels that is most similar to the content of an another image used as reference, the **Template**.

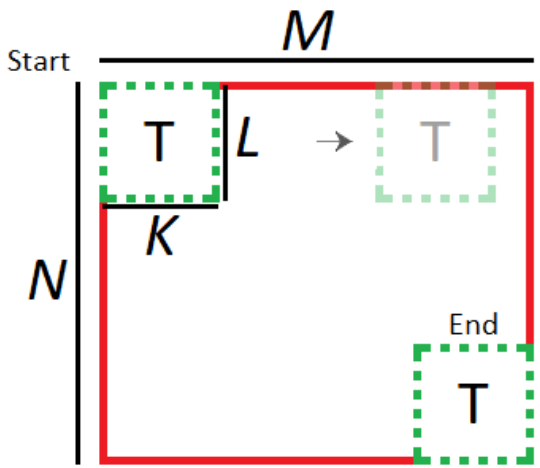


Figure 6: Sliding window procedure. The Template acts as a mask.

Given an input frame $I_{M \times N}$:

- The sub-image $T_{K \times L}$ is superimposed on the first pixel of I and a similarity value of the overlapped region is computed.
- T is moved one pixel at time, always left to the right and top to the bottom.
- At the end of the process, T will have been superimposed $(M - K + 1)(N - L + 1)$ times.

Tracking Algorithms: Template Matching - cont.

- The similarity coefficients are stored in an output matrix. The best match is given by the maximum (minimum) value of the matrix.
- To find out the most suitable metric, we have carried out a comparison test between different matching coefficients.
  - The formula which best suits our goal is the **Zero-mean Normalized Cross-Correlation (ZNCC)** [4].

Formula	Match when still	Accuracy (%)
SQDIFF	yes	54.5
SQDIFF NORMED	yes	52.2
CCORR	no	10.1
CCORR NORMED	yes	97
CCOEFF	no	13.3
CCOEFF NORMED (ZNCC)	yes	100

Figure 7: Similarity metrics comparison.

Tracking Algorithms: Template Matching - cont.

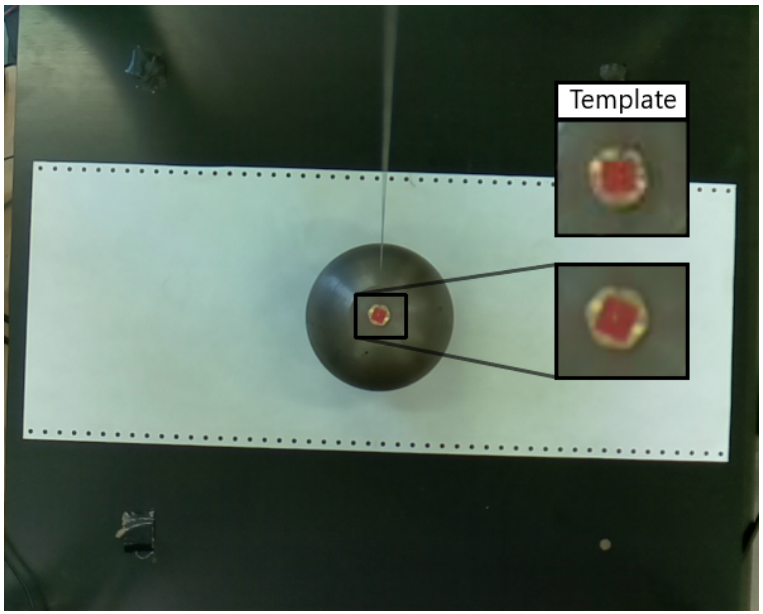


Figure 8: Template Matching applied to a frame.

What to keep in mind:

- The developed program does not implement any kind of pre-processing on the Source image nor to the Template, such as rotation or size normalization. The object to be identified needs to have the **same dimensions and orientation** in both Template and Source.
- The Template holds, besides the marker, a portion of the near area for better performances.
- The high accuracy provided by the calculation of the ZNCC coefficient is however linked to a **high computational cost**, leading to a system that can handle 5 FPS.

Multi-threading

Multi-threading

Multi-threading is the ability of a CPU to execute simultaneously multiple tasks. It can be used to make the most of the available cores.

Multi-threading allowed us to bring the system from 5 to 9 FPS.

Producer-Consumer pattern

The classic **producer-consumer** structure consists of a producer thread that creates something and enqueues it into a buffer. The consumer thread dequeues the element and uses it to perform a task.

Threads involved in the multi-threading solution:

- One **main** thread that extracts frames from the camera.
- Two **computing** threads that consume frames, and produce results.
- A **writer** thread that handles the writing to the CSV file.

Multi-threading - cont.

Between the threads there exists a two **cascading** producer-consumer relation shown below:

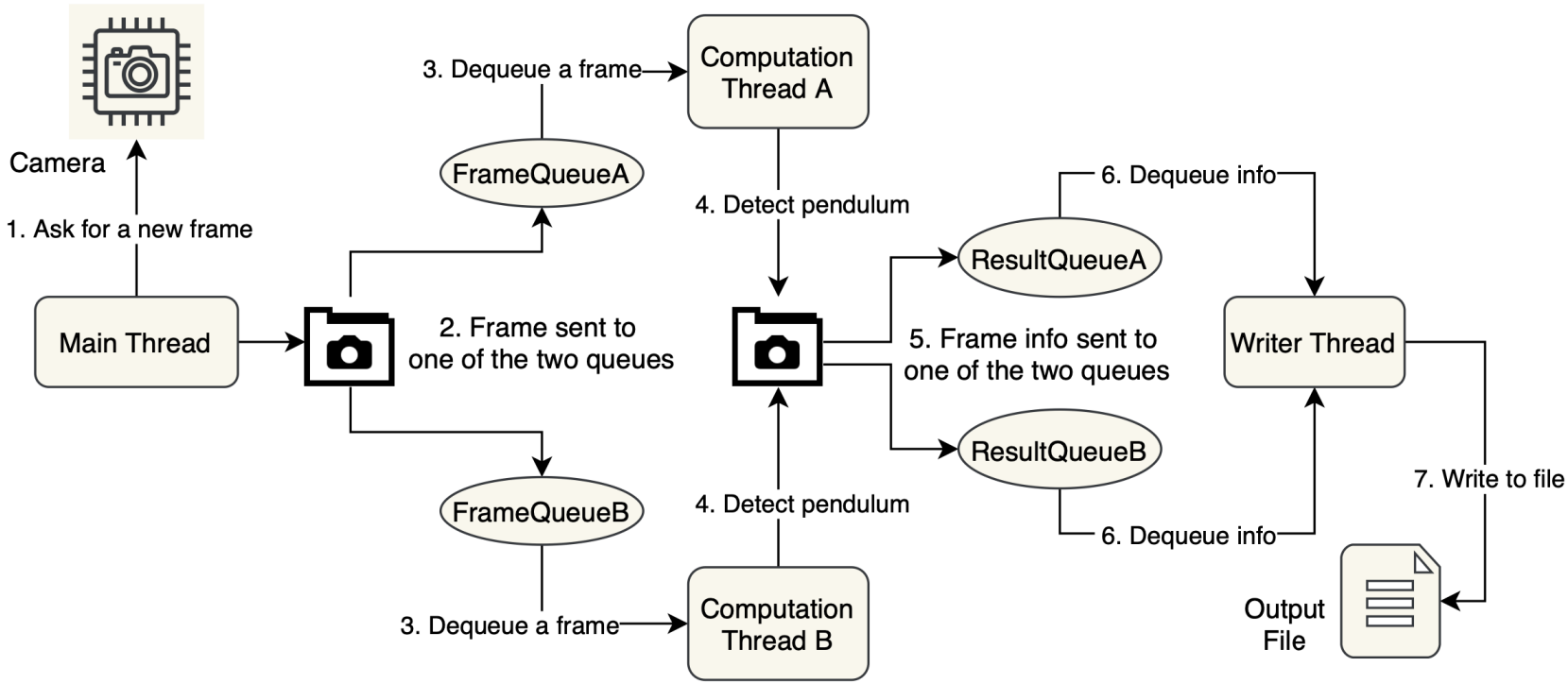


Figure 9: Multi-threading flowchart of threads.

Perspective Correction

Parallax effect

The object position seems different with respect to the background when it is viewed along two different lines of sights [5].

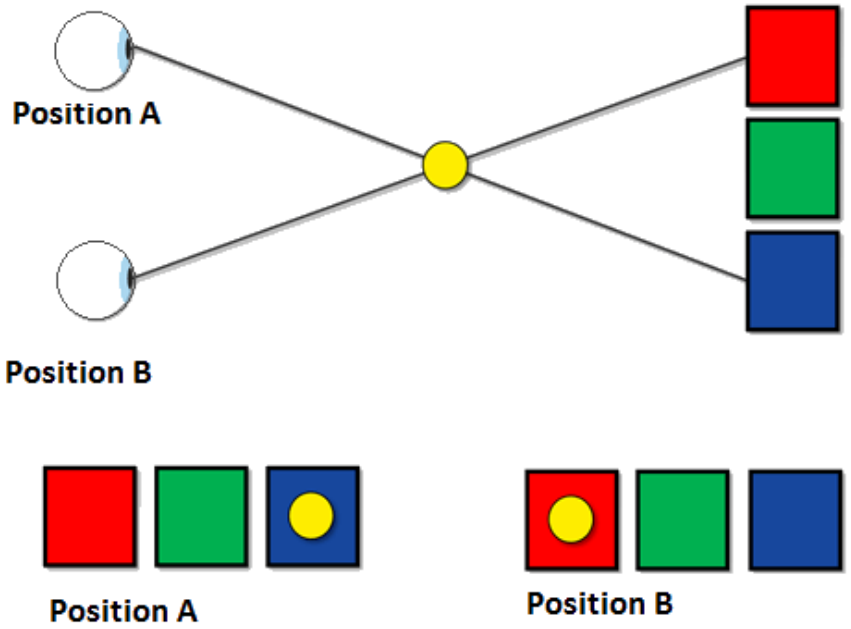


Figure 10: Parallax phenomenon.

Possible solutions:

- Use a calibration plate to calibrate the camera.
- Develop your own perspective correction system.

Perspective Correction - cont.

Idea: apply a perspective transformation to the acquired frames so that it removes the effects of inclination, thus obtaining images projected on the plane perpendicular to the focal axis.

The perspective correction system:

- Is based on Homography **transformation matrix**.
- Allows the user to define its own perspective transformation
  - The user is responsible for placing the points that define the matrix.

Transformation matrix

Given a transformation $T : R^n \rightarrow R^m$ and a point $x \in R^n$, the transformation matrix $M_{n,m}$ associated with T is the matrix such that

$$T(x) = Mx$$

Perspective Correction - Graphical User Interface

- **Trial and error procedure**:
  - The user is responsible for positioning the vertices of a quadrilateral on which to apply the perspective
  - A window shows the warped frame.
- The points are stored in a file and used by the tracking system to perform real-time perspective correction.

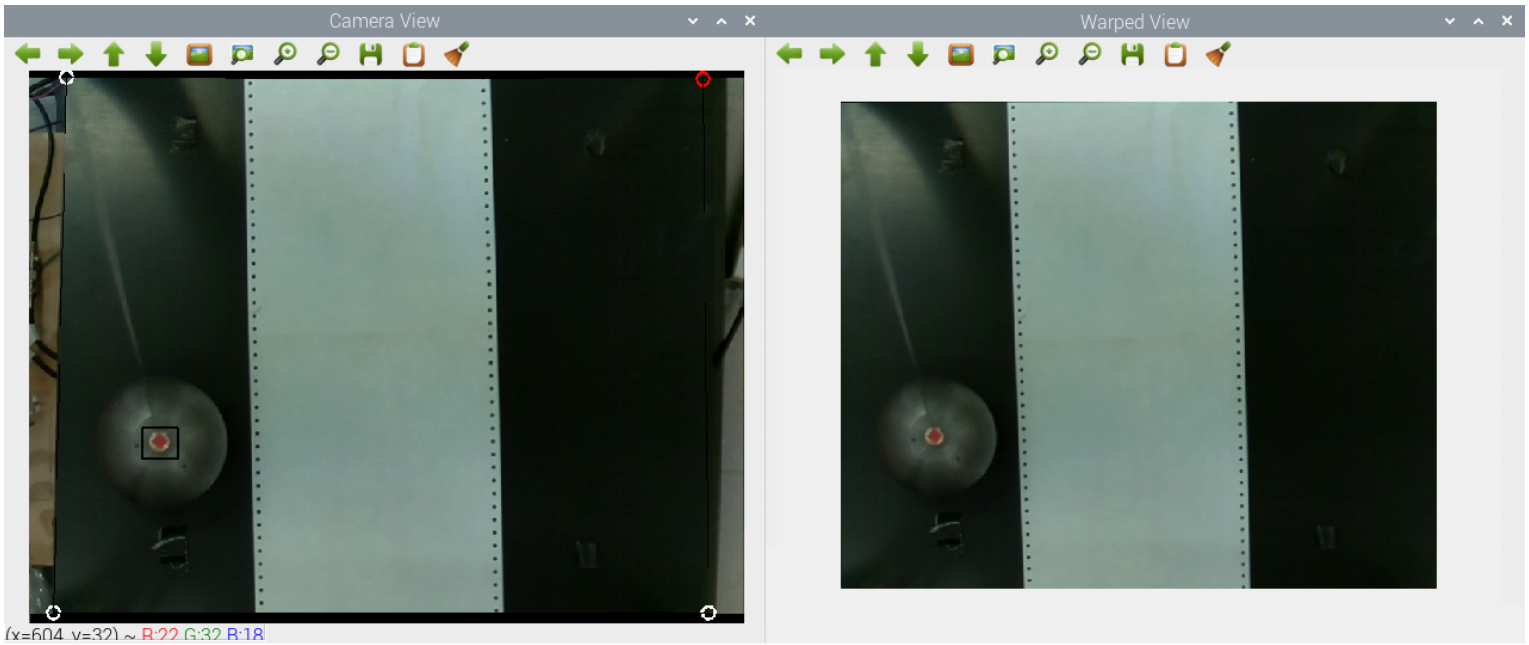


Figure 11: Calibration interface.

Output content

- The coordinates extracted by the system are stored in a CSV file. The entry format is as follows:

`Time;X;Y`

  - `Time`. Expressed in seconds, is relative to the start of the computation. Six decimal places are used to provide a good precision.
  - `X,Y`. Coordinates of the point, expressed in pixels. The axis origin is placed in the top-left corner of the image.

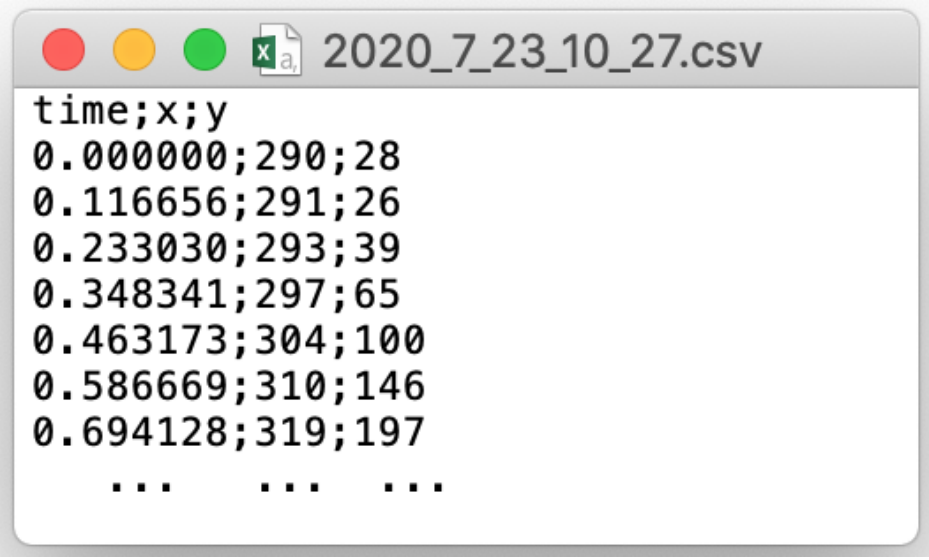


Figure 12: CSV document produced by the tracking system.

Graphical Interface: Tracking interface and 2D Graph

How this method performs:

- The tracking interface allows the user to visualise at a glance the **real-time stream from the camera**.
- The frame shown is the one **before** the perspective correction.
- The image inside the window can be zoomed and saved.

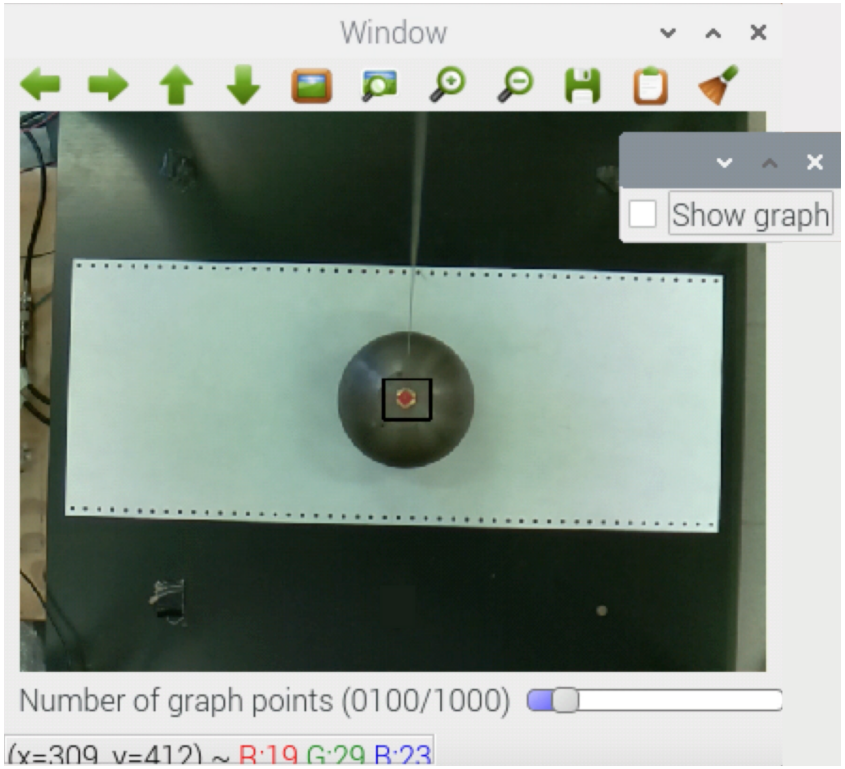


Figure 13: Template Matching window.

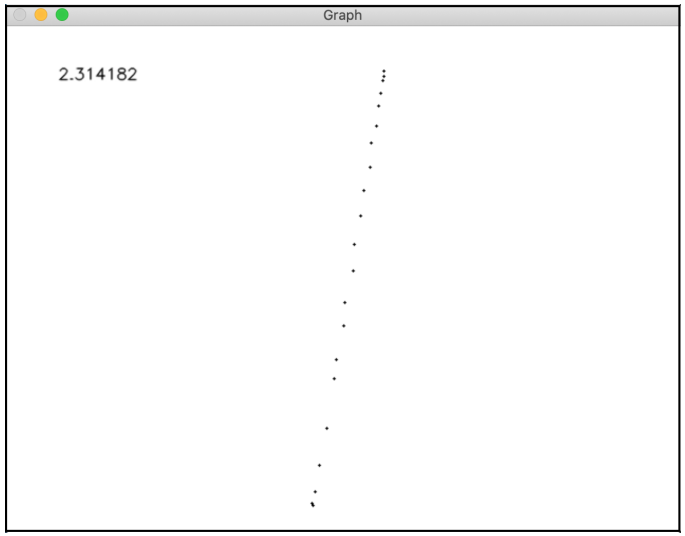
- It can also be shown a **graph of points** detected in real-time (up to 1000). As new points come, the older coordinates are deleted from the graph.

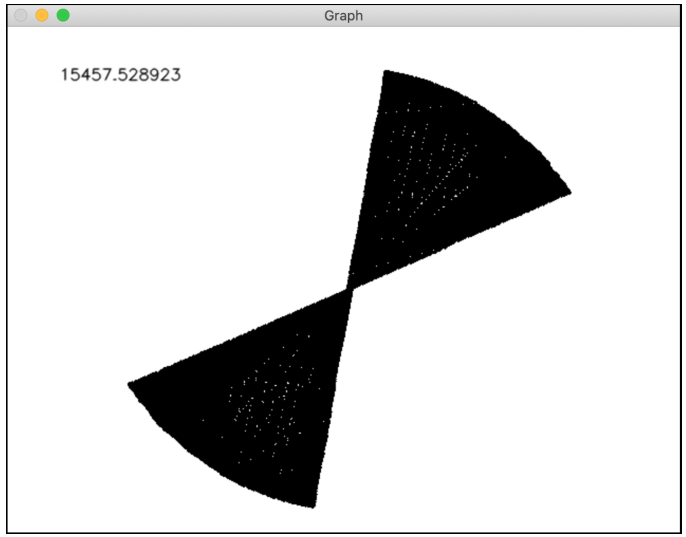
Graphical Interface: 2D Graph, offline version

Similar to the online version, but showing coordinates read from a CSV of choice.

Additional operation:

- The appearing of new coordinates can be **stopped and played**.
- The **speed** of visualisations of new points can be changed.
- The **number of points** displayed at a time can be changed as well (30 by default) with no upper limit.





Conclusions

What was the aim of this project?

To develop a tracking system that could continuously and accurately extract the coordinates of a moving Foucault pendulum, making it possible to analyse its motion.

- **Implemented strategies**
  - Template Matching based on `ZNCC` coefficient computing. It provides maximum precision and noise robustness.
  - **Multi-threading approach**: doubles the number of frames analysed each second.
  - **Manual calibration** which corrects the parallax effect.

Future Works

Possible future developments:

- Investigate the algorithms for computing a fast ZNCC in order to speedup the matching process.
- The tracking system could be flanked by an ideal model simulator in order to juxtapose the real and the expected data
- Use a calibration plate to perform the calibration process and compare the results with the current method.

Thanks! Any questions?

Bibliography

[1] J. Vautherin, S. Rutishauser, K. Schneider-Zapp, H. Choi, V. Chovancova, A. Glass, and C. Strecha, "Photogrammetric accuracy and modeling of rolling shutter cameras," *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. III-3, pp. 139–146, Jun. 2016. DOI: 10.5194/isprsannals-III-3-139-2016.

[2] H. Cheng, X. Jiang, Y. Sun, and J. Wang, "Color image segmentation: Advances and prospects," *Pattern Recognition*, vol. 34, no. 12, pp. 2259–2281, 2001, ISSN: 0031-3203. DOI: https://doi.org/10.1016/S0031-3203(00)00149-7. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0031320300001497.

[3] B. Ribeiro, D. Martins, A. onio, A. Neves, A. Pinho, and D. IEETA, "Arbitrary ball detection using the circular hough transform", Sep. 2020.

[4] *Imaq vision, concepts manual*, http://www.ni.com/pdf/manuals/322916a.pdf, National Instruments Corporation, Austin, Texas, Oct. 2000, pp. 216–217.

Bibliography (cont.)

[5] *Oxford English Dictionary, 3rd edition*. Oxford University Press, 1989.