

**ORIGINAL ARTICLE**

Visualizing Free-falling Motion using Raspberry Pi Microcomputer for Hybrid Teaching Physics Laboratory

*Nabella Holling, Muhammad Afif Abdul Rani, Esther Rachel Beruin and Rafiq Tamin

School of Foundation Studies, University of Technology Sarawak, 96000 Sibul, Sarawak, Malaysia

ABSTRACT - A conventional setup to demonstrate the relationship of height displacement and time for a free-falling object includes the measuring of variables using a second counter and a metre ruler. Such a setup requires the experiment to be repeated for a series of regular, consecutive increments of height while the time for each fall is recorded. To perform the act of measuring height and time in a hybrid setting is not physically possible for remote learning. The setup described in this paper, incorporates the use of object tracking and data logging programmes on the Raspberry Pi microcomputer in place of physical measurements. The improvised setup yields the conventional XY plotter that verifies the relationship between height displacement and time variables, confirming the accuracy and applicability of the setup for a hybrid teaching laboratory.

ARTICLE HISTORY

Received: 13 Dec 2022

Revised : 28 Jan 2023

Accepted: 27 Feb 2023

KEYWORDS

Hybrid teaching laboratory, Raspberry pi, Physics education, Apparatus.

INTRODUCTION

The COVID-19 pandemic has indirectly challenged academics to overcome the physical barriers created by social distancing implemented worldwide. While non-psychomotor subjects could be easily replaced by fully online learning, physics laboratories, require a much more innovative approach. In a physics teaching laboratory, aside from verifying the theoretical concepts learned in lectures, it is in this environment that the students are taught important laboratory techniques such as taking measurements, data acquisition, data visualization, and interpretation. A lab designed to be conducive to hybrid learning must retain these aspects while being physically distant. Options from past works include a computer-based free-fall experiment [1], an Arduino-sensor-based experiment [2], interactive computer simulations [3];[4], and video and multimedia demonstrations [5]. The setup described in [1] is not applicable to long-distance learning because the highlight of the work is the use of the HC-SR04 ultrasonic distance sensor to read the displacement data. Computer simulations, which lack room for error, and passive demonstrations via recorded video do not ensure students' engagement. The use of a Raspberry Pi camera and automated data logger offers an adaptive strategy to renovate the readily available free fall experiment into a setup that is suitable for distant learning.

In a conventional experiment to demonstrate the behaviour of an object that falls under the influence of gravity, students are required to manipulate the height displacements through which the fall takes place. By measuring the time for each height, displacement in terms of time is easily derived from the XY scatter plot of the two mentioned variables.

In a classroom where the emphasis is on data visualisation and interpretation, simulators are widely used. Currently, there are numerous online platforms providing free simulators. Common features in all simulators are the GUI interface to manipulate the heights and the complementary graphs simulated from the defined settings. The graphs to demonstrate the motion of a free-falling object are commonly the height displacement against time and the velocity against time.

MATERIALS AND METHODS

In a hybrid teaching laboratory, instructors need to compensate for the lack of physical measuring tools when performing this experiment. To attend the laboratory session remotely, students must attend the teleconference platform where the session is held. Using the Raspberry Pi camera, a real-time demonstration can be broadcast live to the attendees. The Raspberry Pi camera is improved with an object tracking programme that opens as a "frame" from which the setup can be watched. Using the same 'frame' output, the coordinates of the moving object are registered and given a timestamp. These data are saved as a CSV file at the end of each fall and are downloadable and shareable via online platforms. A free falling experiment is presented using a mathematical term. When the emphasis is on defining factors that participate in a particular phenomenon during a lecture, the intrinsic variables from an accelerating object moving in uniform linear motion allow the derivation of Equation 1:

$$\begin{aligned}
 a &= \frac{dv}{dt} \\
 a \cdot dt &= dv \\
 a \int_0^t dt &= \int_{v_i}^v dv \\
 at &= v - v_i \\
 v &= v_i + at
 \end{aligned} \tag{1}$$

Alternatively, velocity by definition is also given by $v = \frac{dx}{dt}$. Expressing derived quantities using measurable base quantities has the advantage of verifying the equations using physical apparatus. Equation 1 can be further simplified as follow:

$$\begin{aligned}
 v &= v_i + at \\
 \frac{dx}{dt} &= v_i + at \\
 dx &= (v_i + at)dt \\
 \int_{x_i}^x dx &= \int_0^t (v_i + at)dt \\
 \Delta x &= v_i t + \frac{1}{2}at^2
 \end{aligned} \tag{2}$$

Therefore, the physical apparatus used in a general experiment to demonstrate a free-falling object measures the height and the time span of the fall. In this work, the conventional setup uses a semi-autonomous second counter and a metre scale. The microcomputer-based setup measures the height displacement by determining displacements from the Y values of the traced object coordinates. Time displacement is calculated from the timestamp of each recorded position.

Conventional Physics Laboratory Apparatus

The conventional laboratory setup used was obtained from the LD Didactic Laboratory free-fall experiment set and procedure. In the procedure, a metal ball was attached to an electromagnetic holder. The height of the ball was determined prior to release, and a contact plate was placed on the ground surface. The timer began the moment the electromagnetic holder was turned off. The time register stopped the moment the contact plate was triggered. Experiments were repeated at regular height intervals. An XY plot of height against time was generated.

Raspberry Pi Microcomputer-based Laboratory Apparatus

The improvised laboratory setup used was also based on the conventional LD Didactic Laboratory free-fall experiment. However, the metal ball was replaced with a 6 cm-diameter red ball. A metal part was incorporated at the top of the ball to provide a 'sticky' end for the ball to be attached to the electromagnetic holder. Using this setup, the experiment was performed once, using a maximum height of 80 cm. The *cis* terminal (relative to the electromagnetic holder) of the height was tagged using a blue marking, while the *trans* terminal was tagged using a green marking. These markings and the final setup are shown in Figure 1.

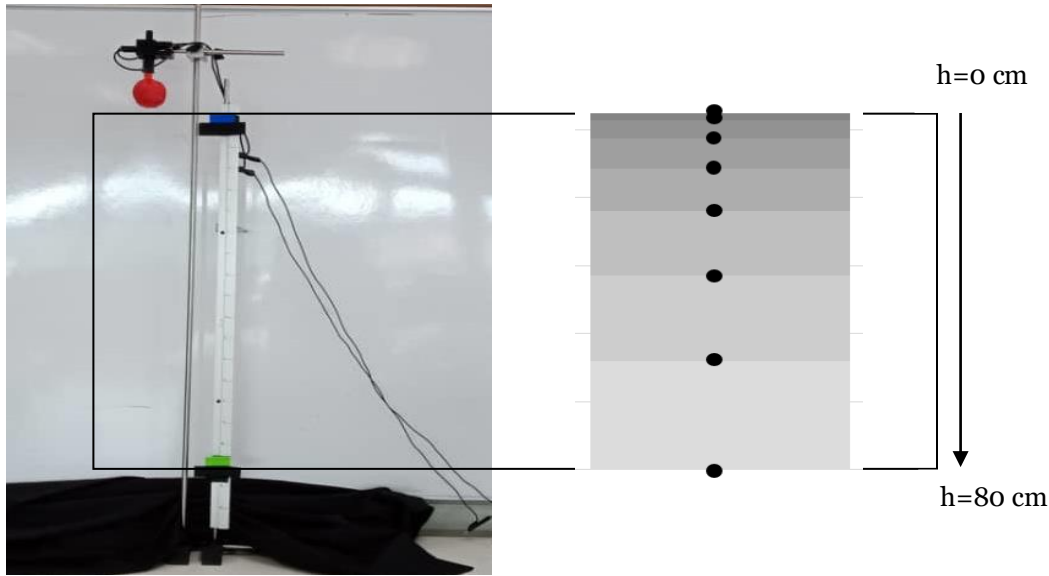


Figure 1. The *cis* and *trans* terminals are marked with a blue and green markings respectively. The inset dot diagram was drawn using data taken from the registered data from the programme.

Next, the designated programme was run. An image frame was shown on the screen when the programme was running. The camera was then adjusted to include red, blue, and green objects that fell within the image frame. Using Thonny IDE, the coordinates of the blue and green markings were used to define `cv2.line` integers and modules. When ready, the ball was released. Data logging took place while the red ball travelled across the blue and green lines shown on the image frame on the screen display. The data was saved independently in a separate folder in CSV format.

Unit Conversion

A 1-metre ruler was positioned parallel to the *y*-axis of the falling motion. From the image frame, the coordinate of the object was noted, and the displacement was obtained using cross multiplication. Referring to Figure 1, given that a one-meter ruler spans across coordinates $Y = 25$ to $Y = 45$, the ratio of meter: pixel becomes 1:20. The mentioned image frame was set to a dimension of 320 x 480 pixels (height x width). This dimension allows the detection of displacements on the vertical and horizontal axes covering an area of 153600 pixel².

Raspberry Pi Microcomputer Programs Overview

The programme was named `dotdiagram.py`. The working overview of the programme is shown in Figure 2. The flowchart shows the commencing of time counter and coordinate registration by the programme once the object fell between the blue-green vertical range. At the declaration section, the modules were listed as shown in Figure 3. The core of this programme is based on OpenCV (Open-Source Computer

Vision Library) [6],[7]. The use of additional import modules was included to enable the timestamping and data logging features of the *dotdiagram.py* program.

The entire time when the frame image was captured, the programme detected red, blue, and green objects using the contour selections command. Under this command, the BGR input was converted to the HSV input. HSV colour profiles were created and saved in *np.array*. The largest contour to generate signals with the defined colour profiles was read as either the red, blue, or green objects and was used by the program. Using if statements, data logging was activated only when the red object coordinates fell on the blue-green range.

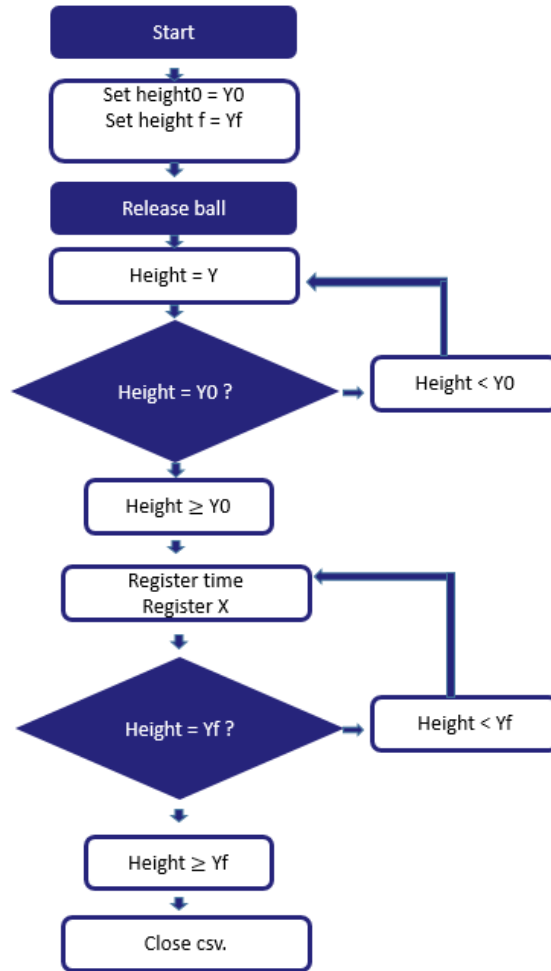


Figure 2. The flowchart of the programme.

```

import cv2
import numpy as np
from time import sleep
import CSV
from datetime import datetime, date
import re
    
```

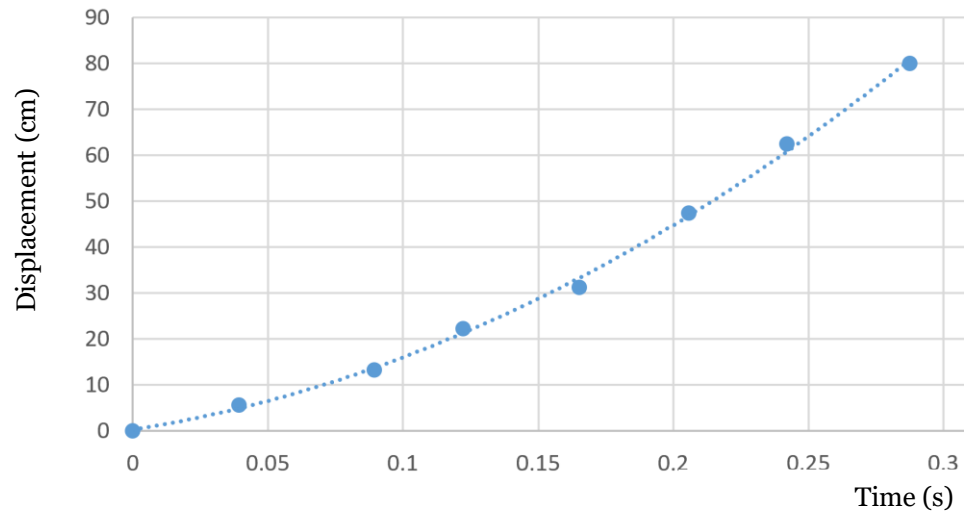
Figure 3. The library declared at the beginning of the programme.

RESULTS AND DISCUSSION

From both setups, the raw data required to generate a scatter plot between displacement and time variables had been successfully carried out. The linearization of curve lines shown in Figure 4 is a crucial step in validating the quadratic function associated with objects under the influence of constant acceleration in uniform linear motion. The best-fit values of the XY scatter plots reflect the precision level of the given setups, and the Raspberry Pi object tracking programme shows better performance relative to setups that rely on repetition and the use of a second counter.

The y-intercept of the graphs indicates zero error. Graph A in Figure 4 indicates time recording was delayed by an amount of a vertical unit, judging from the positive zero error on the displacement axis of the plot. This is an expected outcome because the chances of placing the red object one unit above the blue line, i.e., the moment before release, are limited to the operator's careful precision when positioning the red ball before the blue marking. The interference of signals produced by the red ball and other red objects that fell on the image frame could momentarily alter the actual position of the red ball, resulting in additional gaps between the red ball and its initial position marked by the blue marking.

A.



B.

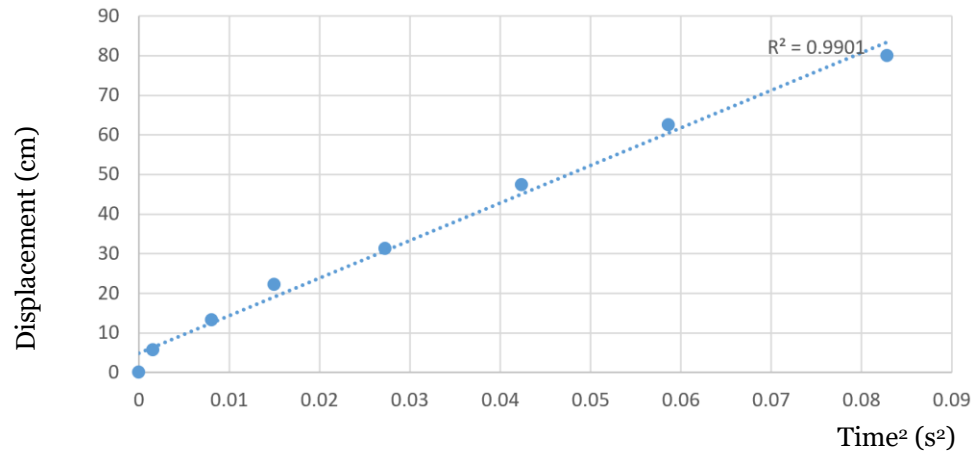


Figure 4. Linearization of displacement and time variable using Raspberry Pi data logger. A. The displacement of fallen object plotted as a time series. B. The effect of squaring the time variable for the distance travelled of the fallen object.

The time ranges differ in both setups owing to the different dimensions of falling objects used in the experiment. The implication of an extended time range can be observed from the gradient of the graphs. The metal ball was used to minimise the effect of unidentified forces other than gravity, while the red ball was used to improve visual and signal detection by the camera. Regardless of experiencing external forces despite gravity's pull, both setups satisfy the condition of having constant acceleration. Further data manipulation from the Raspberry Pi data logger results in the creation of a literal dot diagram of the falling object. This requires additional steps, but an illustration resembling a stop-motion picture can be obtained. The dot diagram shown in Figure 1 was created using a stacked column chart in Microsoft Excel. The time interval between two registered data points was defined at 0.01 second.

In a separate experiment, the *dotdiagram.py* programme was adapted to demonstrate a red ball released from a projectile launcher. The three conditions (if statements) that enabled or disabled the program's data logging feature were replaced with a simple TRUE/FALSE condition of two switch buttons. Even though it is not technically elaborated in this work, the accuracy and precision were notable in the projectile motion experiment. Figure 5A follows the position of the red ball travelling in a projectile motion. The position XY coordinates were used to generate plots of (i) x displacement against time and (ii) y-displacement against time. Linearization using y-displacement data passed at its maximum height is carried out to minimise the zero error of the setup (setup is not shown). The characteristics of the horizontal (x-displacement) or vertical (y-displacement) components of projectile motion were conserved with or without unit conversion and can be presented using the graphs shown (Figure 5A).

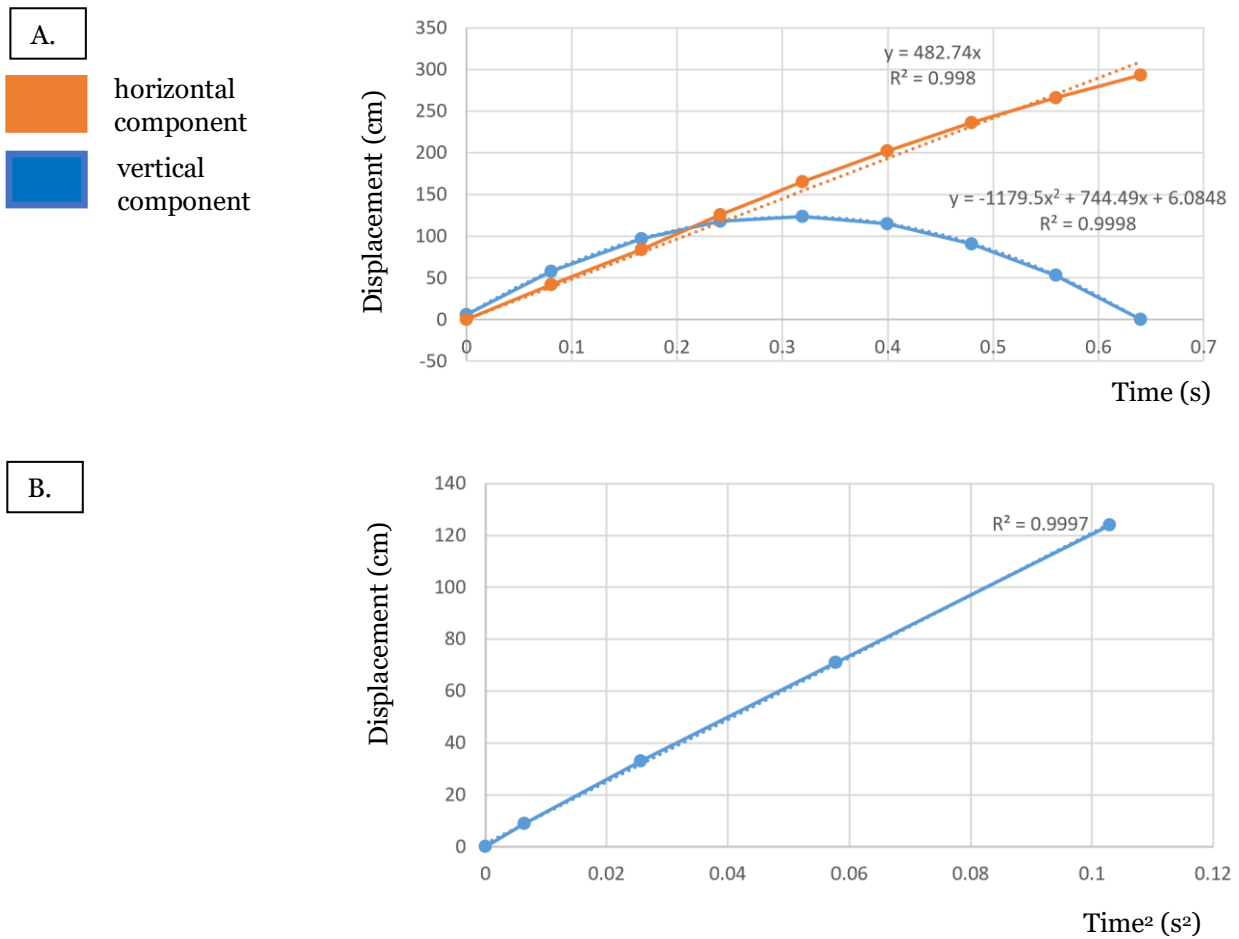


Figure 5. Height displacements starting from maximum height onwards (the last five data points) were selected from A (blue) and linearized as a scatter plot shown in B.

The data shown in this work was initially collected using a CSV file. In the Raspberry Pi program, files with such a format were opened using LibreOffice. This allowed us to demonstrate any data visualisation at the same time and exchange data with other members via online platforms. Even though physical measuring tools were not used or were used at a low level, the microcomputer-based setup allowed the students to spend more time honing their skills in data interpretation and scientific communication.

Though not applied in this work, it has been reported in other studies [8],[9],[10] that full automation and remote access to the Raspberry Pi IDE can be achieved using various Python modules and accessory hardware. In this work, for example, the switch for the electromagnetic holder can be connected to the Raspberry Pi's output pin, allowing the switch to be triggered from a remote location.

By exploring various innovative approaches to deliver classical physics phenomena to students attending the laboratory remotely or face-to-face, the impact of social distancing on the education system will gradually diminish. However, it is important to analyse the strengths and weaknesses of the explored method to determine the extent of its applicability. It is also important to acknowledge the need to train the student operator before the laboratory session. The explanation must include the principles behind the experiment setups, the Raspberry Pi Thonny IDE, and the general workings of the Raspberry Pi camera and video capture.

CONCLUSION

The reported work is thought to enrich the options for academic providers in their methods of teaching delivery, making the learning process less affected by any emerging pandemic or situations requiring social distancing. The experimental setup could be expanded to a national scale in line with state and national efforts to improve STEM education.

ACKNOWLEDGEMENT

The authors thank the University of Technology Sarawak for the opportunity, and support. This study was funded by UTS URG grant UTS/RESEARCH/2/2022/23.

REFERENCES

- [1] Subali, B., Rusdiana, D., Firman, H., Kaniawati, I., & Ellianawati, E. 2017. Computer-based experiment of free fall movement to improve the graphical literacy. *Jurnal Pendidikan IPA Indonesia*, 6(1).
- [2] Moya, A. A. 2018. An Arduino experiment to study free fall at schools. *Physics Education*, 53(5).
- [3] Jimoyiannis, A., & Komis, V. 2001. Computer simulations in physics teaching and learning: a case study on students' understanding of trajectory motion. *Computers & education*, 36(2), 183-204.
- [4] Claessens, T. 2017. Analyzing virtual physics simulations with Tracker. *The Physics Teacher*, 55(9), 558-560.
- [5] Aravind, V. R. 2016. Video and Multimedia in Physics Education. *Engineering and Science Education*, 1(1), 2-2.
- [6] Canu, S. 2019. Control webcam with servo motor and raspberry pi - OpenCV with python [Review of Control webcam with servo motor and raspberry pi - OpenCV with python]. PYSOURCE. <https://pysource.com/2019/06/05/control-webcam-with-servo-motor-and-raspberry-pi-opencv-with-python/>
- [7] Reading from CSV files - Python Programming with Raspberry Pi [Book]. n.d.. Wwww.oreilly.com. Retrieved October 24, 2022, from <https://www.oreilly.com/library/view/python-programming-with/9781786467577/7a6f7f14-ca88-4d49-a9ca-801940838e23.xhtml>
- [8] Jolles, J.W., 2021. Broad-scale applications of the Raspberry Pi: A review and guide for biologists. *Methods in Ecology and Evolution*, 12(9), pp.1562-1579.
- [9] Selvaraj, K., Alagarsamy, S. and Dhilipkumar, M., 2021, May. Raspberry Pi based automatic door control system. *2021 3rd International Conference on Signal Processing and Communication (ICPSC)* pp. 652-656
- [10] Marot, J. and Bourennane, S., 2017, August. Raspberry Pi for image processing education. *2017 25th European Signal Processing Conference (EUSIPCO)* (pp. 2364-2366).