

Estudio de la caída libre utilizando diferentes técnicas experimentales



Montero Germán^{1,2}, García Alfredo², Ríos Víctor², Román Ángel²

¹Departamento de Matemática y Física, Facultad de Humanidades y Educación, Universidad del Zulia. Ciudad Universitaria Avenida Guajira. Maracaibo-Estado Zulia- Venezuela.

²Instituto Universitario de Tecnología de Maracaibo. Av. Principal La Floresta. Maracaibo. Estado Zulia- Venezuela.

E-mail: germanmontero71@gmail.com

(Recibido el 14 de junio de 2017, aceptado el 18 de diciembre 2017)

Resumen

La presente investigación tiene como propósito, utilizar varias herramientas tecnológicas para estudiar el movimiento de caída libre de los cuerpos con datos actuales obtenidos mediante el programa analizador de video Tracker 4.87, una calculadora científica Casio 300 con sus respectivos sensores y el programa Python 3.4.4, para que posteriormente se comparen los resultados estos métodos. Los datos experimentales del movimiento de caída libre se obtuvieron mediante la utilización de un balón de Fútbol colocado a una cierta altura en el laboratorio. La técnica consistió en procesar los datos experimentales obtenidos por un sensor de la PASCO conectado a la calculadora Casio ClassPad 330, procesarlos y simularlos mediante los software libre: "Tracker" y "Python". En esta experiencia se pudo observar la versatilidad en la utilización de los programas mencionados, ya que pueden servir como herramienta didáctica innovadora y de fácil acceso para la enseñanza de la física.

Palabras clave: Caída libre, Video Tracker, Software Python para física, Casio-Sensor en enseñanza de la física, Instrumentación tecnológica en física.

Abstract

This research aims to use several technological tools to study the movement of free falling bodies with real data obtained using an analyzer video program called Tracker version 4.87, a scientific calculator Casio 300 with their respective sensors and the Python software version 3.4.4, to later compare the results of these methods. Free falling movement experimental data were obtained using a soccer ball placed at a certain height in the laboratory. The technique consisted of processing the experimental data obtained by a sensor connected to Pasco ClassPad 330 Casio calculator, process and simulate using the free software "Tracker" and "Python". In this experience, it could be observed versatility in the use of these programs, as they can serve as innovative and easy access tools for teaching physics.

Keywords: Free falling, Video Tracker, Software Python for physics, Casio -Sensor in teaching physics, physics instrumentation technology.

PACS: 07.05.Hd, 07.05.Tp, 07.05.Rm

ISSN 1870-9095

I. INTRODUCCIÓN

La educación actualmente está atravesando problemas críticos a nivel social, económico y político, se piensa que la enseñanza de la física, se debe adaptar a este nuevo tiempo, ya que el estudio de la misma es de suma importancia por ser la ciencia que da respuesta a los fenómenos que ocurren en la naturaleza. Una de las formas que permite que fluya esta ciencia es estudiando algunos métodos para hacer más económicos y fácil de manejar los problemas experimentales de la física y en este caso lo que se pretende investigar es el movimiento de la caída libre de los cuerpos, mediante sensores de movimiento tecnológicamente adaptados y en la utilización de software libre con capacidades de realizar aplicaciones científicas.

En los cursos de física mecánica y en particular, en aquellos de mecánica clásica, se estudia con normalidad el

movimiento de los cuerpos en una y dos dimensiones. Tal es el caso del movimiento vertical de caída libre que se experimenta de manera sencilla e ideal sin considerar algunos factores que influyen a dicho movimiento.

En el estudio de la cinemática mediante el uso del software Modellus, se observa detalladamente el problema de la caída libre, según Sánchez [1], indica que el uso del software puede contribuir a que los estudiantes se apropien mejor de las bases de la mecánica newtoniana, ya que, existen aspectos que ayudan a comprender este tipo fenómeno cuando se observan varios casos. Este trabajo da la oportunidad de presentar cambios de parámetros, de manera que los estudiantes se incentiven y se destaquen por su participación en las actividades propuesta por el profesor.

Gracias a las nuevas tecnologías, se pueden estudiar los fenómenos físicos de diferentes formas. Un ejemplo de ello

es el que propone Ríos [2], lo cual destaca que el uso de simulaciones ayuda a comprender algunos temas de la física elemental, de forma atractiva y novedosa. Así como también Begeler [3], obtiene resultados favorables que relacionan los parámetros de roce en la influencia del movimiento de caída libre. Este mismo autor utiliza sensores de la Casio – Pasco, para adquirir la data de este tipo de movimiento.

Por otro lado, para la realización de este estudio, se utilizan herramientas tecnológicas que implican aquellas propuestas por Begeler [3] y Montero[4] para la adquisición de datos utilizando el sensor Casio-Pasco, la propuesta por Ríos [2] que utiliza el software libre Tracker en su versión 4.87, y las simulaciones de la caída libre desarrolladas con el software libre Python versión 3.4 en los Laboratorios de Física del Instituto Universitario de Tecnología de Maracaibo, con la cual se obtendrá una aproximación del comportamiento fenomenológico. El propósito final, es determinar cuál de estas herramientas puede ser la más adecuada, asequible y confiable al momento de impartir en el proceso de enseñanza y aprendizaje de la física, la caída libre de los cuerpos. La teoría utilizada en este estudio se presenta en los textos de física [5, 6, 7].

II. PROCEDIMIENTO EXPERIMENTAL

La experiencia planteada en esta investigación, emplea una instrumentación tecnológica que está compuesta por: una Calculadora CASIO CLASS PAD 330, una Cámara para capturar el video del movimiento del objeto en caída libre, el software Tracker versión 4.87 para procesar el video y para la simulación del movimiento, el software Python 3.4.4 con sus respectivas librerías.

El experimento consistió en mover una pelota de fútbol hacia el suelo desde cierta altura con dos condiciones iniciales: en la primera se dejó caer desde una altura de 2,80 m y la segunda se lanzó con una velocidad inicial (Ver figuras 1 y 2). El procedimiento de cómo se utilizaron los instrumentos y se realizaron los cálculos se explican en las siguientes etapas:

A. Etapa I: Uso de la calculadora y sensor

La Calculadora Casio Class PAD 330, está diseñada para medir y registrar cantidades físicas tales como: temperatura, ondas sonoras, diferencia de potencial, movimientos de objetos entre otros. En este caso se va a registrar el movimiento del balón y en particular se conecta de forma adecuada un sensor marca Pasco provisto de una interface compatible con la calculadora y ésta a su vez se conecta a un Data Analyzer EA-200 marca Casio, tal como se muestra en la figura 1. El programa interno de la calculadora, denominado *E-ConEA200*, registra la altura para cada tiempo del balón de fútbol en caída libre, que posteriormente se traza la gráfica de la altura versus tiempo. El experimento se realizó tomando una altura de 280 cm, medido desde el piso, y se consideró el lanzamiento separado del sensor de movimiento 60 cm, ver figura 1 y 2,

tal como lo indica la documentación descrita en el manual del sensor Pasco. Lo ventajoso del procedimiento con esta instrumentación, es que se tiene de forma inmediata la adquisición de los datos, tanto de la posición como del tiempo en la caída libre del objeto, para su posterior análisis gráfico computarizado mediante diferentes programas como por ejemplo: el Origin [www.originlab.com][8] , el MatLab[9], Mathematica [10] entre otros.

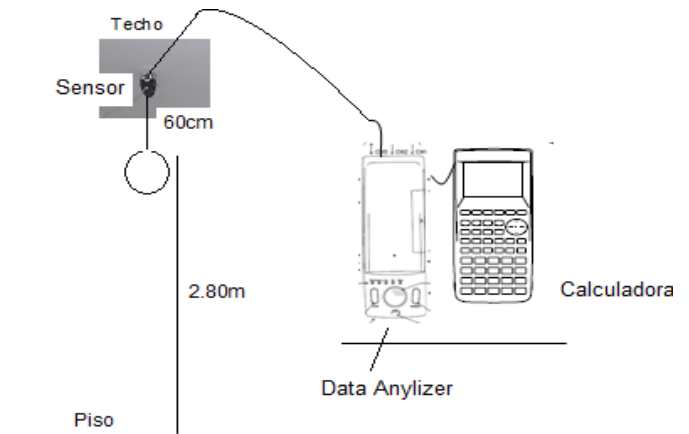


FIGURA 1. Montaje experimental con el uso de la calculadora-sensor.

B. Etapa II: Uso del Tracker

En este caso, al mismo tiempo que se realizaba la experiencia con el uso del sensor, el analizador de datos y la calculadora, se capturaron videos del balón de fútbol en caída libre, sin y con velocidad inicial, mediante una cámara digital de 14.0 Mega-pixels, Marca SANYO modelo VPC-T1496. Después de la grabación, se procedió a analizar el video utilizando el software Tracker versión 4.87, así pues se logró digitalizar el movimiento de la caída libre del objeto. Este software computarizado dispone de una función avanzada que permite realizar el seguimiento del objeto, denominada Auto-Tracker, además de registrar su posición y su tiempo de manera secuencial sólo pulsando clic con el ratón conectado a la computadora, en nuestro caso una PC de medianas prestaciones. Una vez capturados los datos del video, se grafican con el mismo programa y se analiza el comportamiento de la posición en función del tiempo. Se destaca la posibilidad de este software analizador de videos de disponer de la data resultante para ser utilizada con otros programas de capacidades graficas más avanzados como los mencionados en la Etapa I de este estudio. Después en el bucle *for* de las líneas de la 8 a la 11 se lee línea a línea los datos separados por una coma (,) tal como se hace en la línea 9. El resultado es que se almacena el tiempo y la altura en las variables $x1$ y $y1$ respectivamente. Al terminarse la lectura del archivo, finaliza el bucle *for* y en la línea 12 se cierra el archivo. En la línea 13 se almacena en la variable *sol* (solución) el ajuste cuadrático de los datos almacenados en el archivo. En la línea 15 se imprime en pantalla la solución del ajuste.



FIGURA 2. Ensayo del lanzamiento en caída libre de un balón de Fútbol. Toma de video y fotografía.

C. Etapa III: Simulando la caída libre del objeto con el software Python

La modelación experimental que se desarrolló utilizando el software Python con algunas de sus librerías, generar la gráfica de la altura versus el tiempo del movimiento en caída libre del balón de fútbol con su respectivo ajuste polinomial cuadrático y la simulación de los ensayos realizados anteriormente. En este caso se tomaron en cuenta los datos adquiridos por el sensor de movimiento a fin de calcular el valor de la aceleración de la gravedad terrestre. Por consiguiente se presentan a continuación una breve descripción de los scripts correspondientes realizados con este software libre.

D. Descripción del algoritmo de cálculos y gráficas

En la línea 2 del código, se activa la librería *matplotlib* para que estén disponibles las funciones gráficas. En la 3, se importa la librería *numpy*. En la siguiente línea se activa de la librería *scipy* el ajuste polinomial el cual se va a utilizar en el ajuste cuadrático de los datos. Luego, en la línea 5 se abre el archivo que contiene los datos de tiempo y altura que alcanza el objeto en caída libre. En las líneas 6 y 7 se inicializan los arreglos que almacenan los datos del tiempo (*x1*), y la posición del objeto en caída libre (*y1*), cuadrático que regresa la función *polyfit()* de la librería *scipy* y en las líneas de la 16 a la 18 se imprime los valores de *a*, *b* y *c* de la ecuación: ax^2+bx+c .

Ahora bien, en las líneas de código de la 22 a la 26 se define la función que grafica la función con los valores del

Estudio de la caída libre utilizando diferentes técnicas experimentales ajuste cuadrático. En la línea 27 se utiliza *linespace()* de la librería *numpy* con la finalidad de suministrar datos entre un valor inicial y un valor final, en este caso se tomó 0.3 que es menor que el menor tiempo del archivo que contiene los datos y 1.1 que es el mayor de los tiempos en el mismo archivo.

En las líneas de la 29 a la 33 se muestra en el gráfico, el título con: *plt.title()*, la etiqueta del eje *x* con: *plt.xlabel()*, y la etiqueta del eje *y* con: *ylabel()*. Seguidamente, se activa la rejilla con *plt.grid()* y finalmente se enciende la visualización de varios gráficos en uno sólo con *plt.hold(True)*. Todas estas funciones de la librería *matplotlib*.

La gráfica de los datos ajustados se realiza con la instrucción de la línea 34. Y en las líneas de 36 a la 41 se muestran todos los datos, tal como están en el archivo de datos. En la línea 42 se utiliza la instrucción *plt.show()*, de *matplotlib*, con la finalidad de que se muestren los gráficos (ver cuadro 1).

Cuadro 1

Código en Python 3.4.4
1 # Regresión cuadrática para la experiencia de Caída Libre
2 import matplotlib.pyplot as plt
3 import numpy as np
4 from scipy import polyfit # Activa de la libreria scipy la función ajuste polinomial
5 mi_archivo = open("caida1.txt") # Abre el archivo caida1.txt que contiene los datos
6 x1=[]
7 y1=[]
8 for line in mi_archivo:
9 x, y = line.split(',')
10 x1+=[float(x)]
11 y1+=[float(y)]
12 mi_archivo.close()
13 sol=polyfit(x1,y1,2) # Agui se calculan los coeficientes del ajuste cuadrático
14 print("La ecuación de la regresión cuadrática es: ax^2+bx+c ")
15 print("Sol=",sol)
16 print("a=",sol[0])
17 print("b=",sol[1])
18 print("c=",sol[2])
19 print("Los datos son:")
20 print("x1=",x1)
21 print("y1=",y1)
22 def f(x2): # Función que devuelve el valor de la altura con los datos del ajuste
23 a=sol[0]
24 b=sol[1]
25 c=sol[2]
26 return a*x2**2+b*x2+c
27 x2=np.linspace(0.3,1.1)
28 # Se traza la gráfica correspondiente a la ecuación de los datos ajustados
29 plt.title("Gráfica de Altura vs tiempo para caída libre")
30 plt.xlabel("Tiempo (s)")
31 plt.ylabel("Altura (m)")
32 plt.grid()
33 plt.hold(True)
34 plt.plot(x2,f(x2))
35 # Se grafican los datos del archivo
36 i=0

```

37 num_ele=np.size(x1)
38 print("Num Ele=",num_ele)
39 while i<=num_ele:
40     plt.plot([x1],[y1],"rv")
41     i+=1
42 plt.show()

```

E. Descripción del algoritmo de simulación

1. Se deben aperturar las librerías necesarias para comenzar el programa. Para este caso se utilizará el `matplotlib` en la línea 2
2. En la línea 3 del código, se activa la librería `matplotlib` para que estén disponible las funciones gráficas y se guarda en el alias `plt`.
3. Luego, en la línea 4 se abre el archivo que contiene los datos de tiempo y altura que alcanza el objeto en caída libre. Seguidamente, en las línea 5 y 6 se inicializan los arreglos donde se almacenarán los datos del archivo: en `x1` se almacenarán los tiempos y en `y1` las posiciones en el “eje `y`” del objeto en caída libre.
4. Después, en las líneas de la 7 a la 10 de código Python se ejecuta un bucle `for` que lee línea a línea los datos en el archivo los cuales están separados por una coma (,), ver línea 8. El resultado de este bucle es almacenar el tiempo y la altura en las variables `x1` y `y1` respectivamente. Al terminarse la lectura del archivo, es decir que no se consiguen más datos, se sale del bucle `for` y en la línea 11 se cierra el archivo.
5. En las líneas de la 12 a la 16 se inicializa el contador del bucle `while`, en la 13 se asigna a la variable `alfa` el número de datos que hay dentro del arreglo de alturas `y1` []. En las líneas 14, 15 y 16 se dibuja el rastro de caída libre del objeto utilizando la función `scatter` de la librería `matplotlib` con todos los datos. Es notable destacar que aquí se mantiene la posición en el eje `x=0` y el signo en el arreglo de `y1` [], se debe a que el objeto se mueve hacia abajo. En las líneas de la 17 a la 19 se definen las etiquetas del gráfico y finalmente en la línea 20 se muestra el arreglo de tiempos y en la 21 el de alturas. (ver cuadro 2).

Cuadro 2

Código en Python 3.4.4. de gráfica de un objeto en caída libre

```

1 # Programa que grafica la posición de un objeto en caída libre
2 #
3 import matplotlib.pyplot as plt
4 mi_archivo = open("caida1.txt")
5 x1=[]
6 y1=[]
7 for line in mi_archivo:
8     x, y = line.split(',')
9     x1+=[float(x)]
10    y1+=[float(y)]
11 mi_archivo.close()
12 i=0 # Se inicializa i
13 alfa=len(y1)
14 while (i<alfa):
15     plt.scatter(0,-y1[i])
16     i+=1
17 plt.title("Gráfica de Altura vs tiempo para un objeto en caída
18 libre")
19 plt.ylabel("Altura (m)")
20 plt.xlabel("Posición en el eje x")
21 print ("t=",x1)
22 print ("y=",y1)
23 plt.show()

```

III. RESULTADOS Y ANÁLISIS

Del estudio de la caída libre, realizado con la calculadora Casio y el sensor, el procesador de video Tracker y el programa realizado en Python, se obtuvieron graficas de la posición del balón de fútbol en función del tiempo. En la figura 3A se muestra el comportamiento del movimiento en caída libre, arrojado por la calculadora. En la figura 3B se observa un comportamiento cuadrático del movimiento rectilíneo uniformemente acelerado, obteniendo un valor de la gravedad experimental de 9.62736 m/s^2 . Este valor comparado con la gravedad estándar de 9.806 m/s^2 [11], se produce un error experimental de 1.85%.

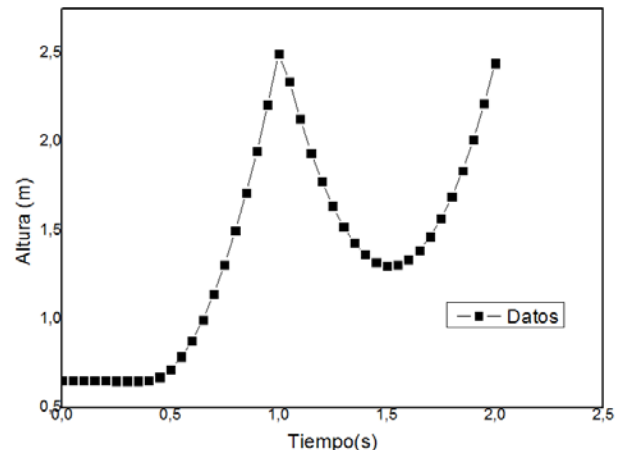


FIGURA 3. A) Se muestra la grafica con datos completos.

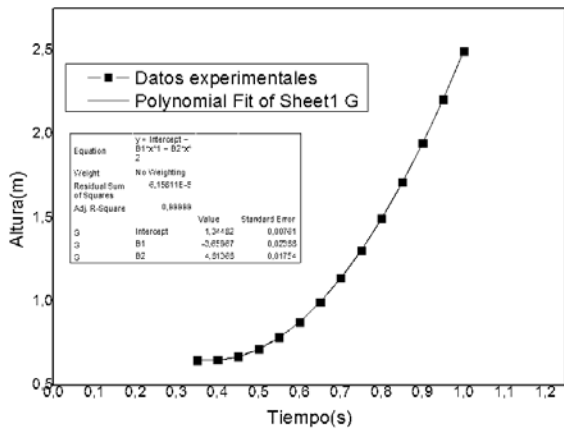


FIGURA 3. B) Gráfica con datos analizados. Ajuste polinomial.

El ensayo realizado con velocidad inicial v_0 , se observa en las figuras 4A y 4B. El comportamiento del objeto en caída libre es similar al anterior. Por otro lado, en la figura 4 A, se muestran todos los datos de la experiencia y en la figura 4 B, la gráfica de los datos analizados. Con esta última se obtuvo un valor de la gravedad experimental de 9.82992 m/s^2 , es decir, como $(\frac{1}{2})gt^2 = (B2)t^2$, entonces $g = 2(B2) = 2(4,91496) = 9.82992 \text{ m/s}^2$, que al compararla con la gravedad aproximada de 9.806 m/s^2 se tiene un error experimental de 0.23%. El valor de B2 se muestra en la tabla 1, además de otros resultados del ajuste cuadrático realizado con el programa Origin versión 8.0 de los valores obtenidos.

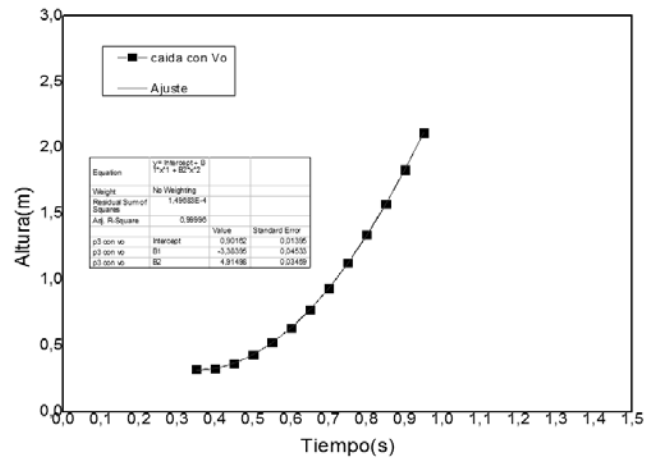


FIGURA 4 B. Gráfica con datos analizados. Ajuste polinomial.

TABLA I. Resultados obtenidos del ajuste cuadrático realizado en Original.

Equation	y = Intercept + B1*x^1 + B2*x^2		
Weight	No Weighting		
Residual Sum of Squares	6.15811E-5		
Adj. R-Square	0.99999		
	Value	Standard Error	
G	Intercept	1,34482	0,00761
G	B1	-3,65967	0,02388
G	B2	4,81368	0,01754

A) Resultados con $v_0=0$

En La figura 5, se muestra el comportamiento del movimiento de la caída libre simulado con el programa Tracker. Los resultados obtenidos muestran que el balón tuvo un movimiento rectilíneo uniformemente acelerado, obteniéndose un valor de la gravedad de 9.9934 m/s^2 con velocidad inicial 0 m/s .

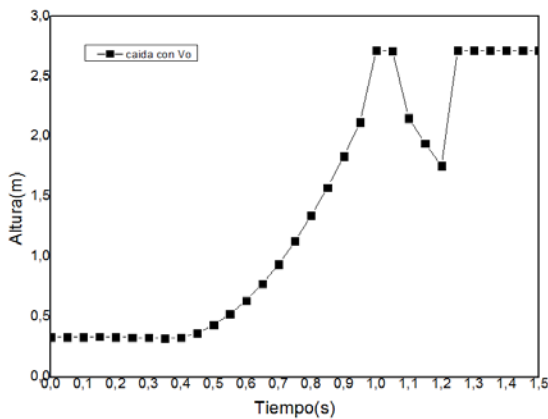


FIGURA 4A. Se muestra la gráfica con datos completos.

Equation	y = Intercept + B1*x^1 + B2*x^2		
Weight	No Weighting		
Residual Sum of Squares	1.49683E-4		
Adj. R-Square	0.99999		
	Value	Standard Error	
p3 con v0	Intercept	0.90182	0.01396
p3 con v0	B1	-3.38396	0.04533
p3 con v0	B2	4.91496	0.01754

B) Resultados con v_0 .

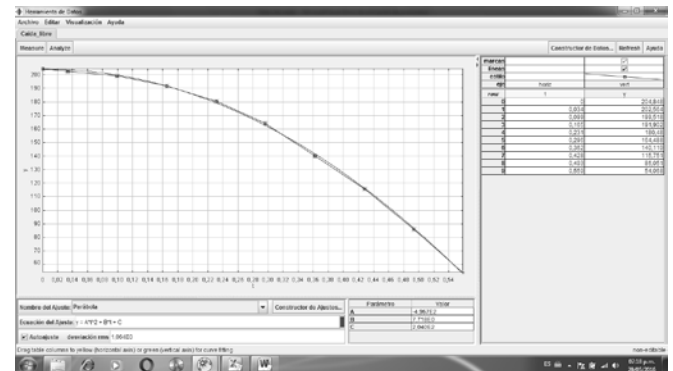


FIGURA 5. Gráfica experimental Tracker de la posición en función del tiempo con velocidad inicial=0.

Montero Germán

La comparación entre la gravedad experimental de 9.934 m/s² con la gravedad aproximada de 9.806 m/s², arrojo un error experimental de 1.30 %.

En La figura 6, se muestra el comportamiento del movimiento de la caída libre simulado con el programa Tracker. Los resultados obtenidos muestran que el balón también posee movimiento rectilíneo uniformemente acelerado, obteniéndose un valor de la gravedad de 9.886 m/s² con velocidad inicial 0.3461 m/s es decir de 34,61 cm/s. Para este caso, la comparación entre la gravedad experimental de 9.886 m/s² con la gravedad aproximada de 9.806 m/s², arrojó un error experimental de 0.80 %.

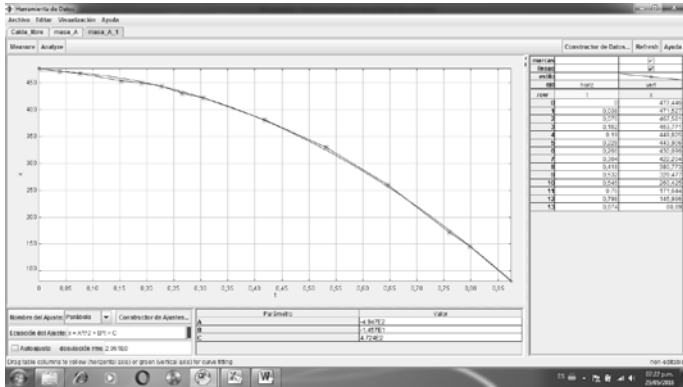


FIGURA 6. Gráfica sobre el comportamiento del movimiento de la caída libre simulado con el programa Tracker.

En el mismo sentido, los resultados obtenidos con Python se muestran en las figuras 7 para el movimiento en caída libre $v_0=0$ y la figura 8 para el movimiento vertical con velocidad inicial probable. En este caso se puede observar que el comportamiento de los dos movimientos es semejante a los obtenidos con las técnicas anteriormente explicadas.

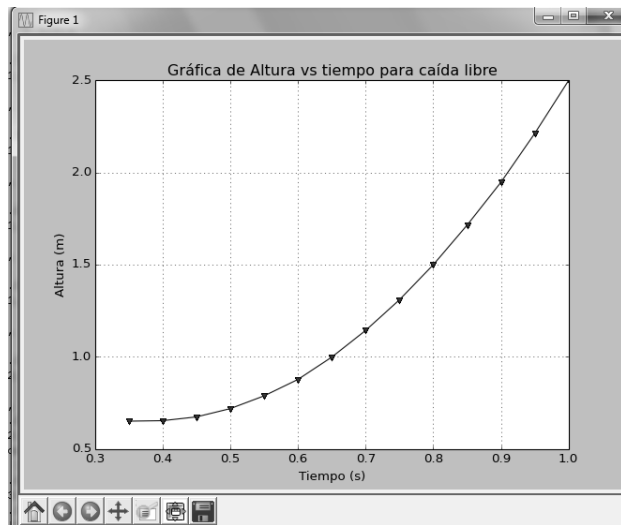


FIGURA 7. Gráfica obtenida del código Python correspondiente a la caída libre $v_0=0$, Datos experimentales y ajuste cuadrático.

Resultado experimentales arrojados por el ajuste cuadrático
 La ecuación de la regresión cuadrática es: ax^2+bx+c
 Sol= [4.81367033 -3.65965626 1.34481765]
 $a= 4.81367032967$
 $b= -3.65965626374$
 $c= 1.34481764835$

Estos valores dan como resultado un valor de la gravedad de 9,627 m/s² teniendo como error experimental con la gravedad promedio de 9,806 m/s² un valor de 1.825%, lo cual se considera muy bajo y aceptable para este procedimiento.

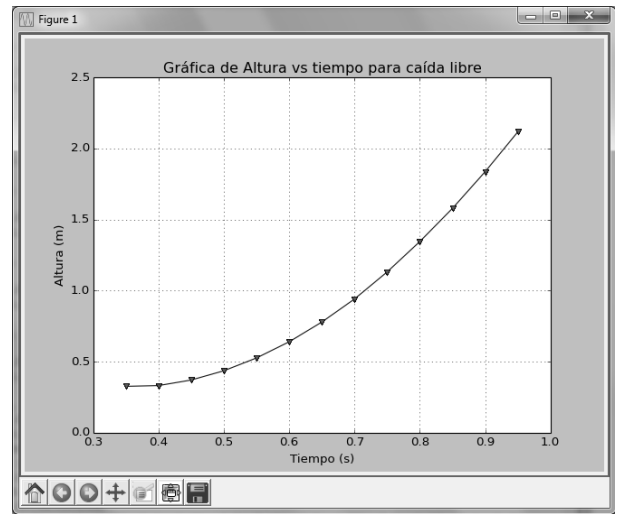


FIGURA 8. Gráfica obtenida del código Python correspondiente a la caída libre v_0 =inicial, datos experimentales y ajuste cuadrático.

Resultado experimentales arrojados por el ajuste cuadrático
 La ecuación de la regresión cuadrática es: ax^2+bx+c
 Sol= [4.91498362 -3.38395442 0.9016239]
 $a= 4.91498361638$
 $b= -3.38395441558$
 $c= 0.901623904096$

Estos valores dan como resultado una magnitud de gravedad de 9,829 m/s² teniendo como error experimental con la gravedad promedio de 9,806 m/s² con un error porcentual de 0.234%.

Por otro lado, se observa la simulación tipo fotográfica (de las exposiciones) del comportamiento en caída libre y con velocidad inicial del balón de futbol. En la figuras 9 y 10 se muestra tales comportamientos, donde se ven más separados, al inicio del movimiento, los puntos cuando existe un impulso de velocidad inicial que incrementa la separación de desplazamientos entre los intervalos de tiempos iguales. Esta simulación tipo fotografía instantánea del movimiento en caída libre, lo realiza la función scatter de las librerías del matplotlib.

Se analiza que para los métodos experimentales e informáticos, utilizados en esta investigación, arrojaron

errores muy bajos, confirmando la eficiencia para la adquisición y procesamiento de datos de estos métodos.

Asimismo, la ventaja que tiene el programa Tracker, es que tiene la virtud de hacer sencilla la adquisición de los parámetros del movimiento en caída libre, por ejemplo se procesa en el mismo programa y se obtiene un ajuste cuadrático, cuyos resultados son excelentes. La comparación de resultados se presenta en la tabla II.

TABLA II. Comparación de resultados de los tres métodos.

Técnicas	Casio-Sensores	Video Tracker	Python
$v_0=0$	9.62736 m/s ²	9.934 m/s ²	9,627m/s ²
Velocidad inicial	9.82992 m/s ²	9.886 m/s ²	9,829m/s ²

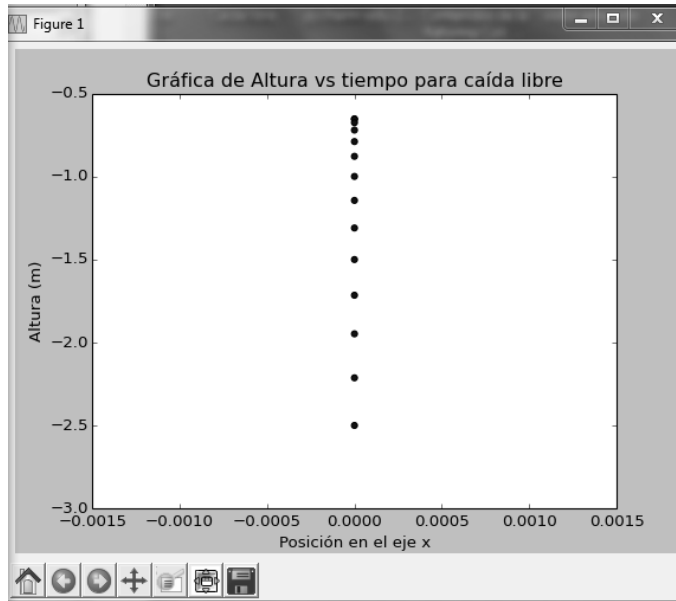


FIGURA 9. Diferentes posiciones del objeto en caída libre, utilizando la simulación con Python.

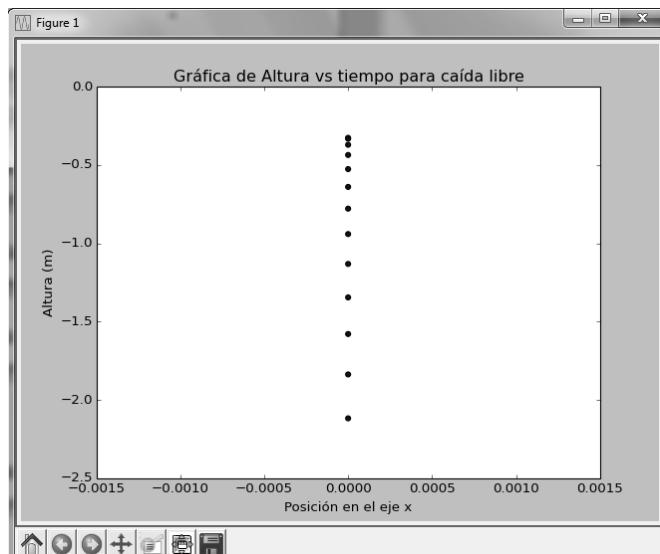


FIGURA 10. Resultados de diferentes posiciones del objeto con velocidad inicial utilizando la simulación con Python.

IV. CONCLUSIONES

En esta investigación se logró comprobar experimentalmente el comportamiento del movimiento en caída libre de un balón de futbol, utilizando tres formas de obtener los resultados. Se concluye que para los métodos aplicados los resultados fueron satisfactorios, obteniéndose errores experimentales muy bajos mediante el cálculo de los parámetros asociados, utilizando el sensor de movimiento y una cámara digital para capturar el video de movimiento presentado. Finalmente los datos fueron modelados de forma adecuada utilizando los softwares Tracker y Python

Un análisis comparativo, se realiza al saber que la calculadora Casio con sus sensor es más versátil para la adquisición de los datos y la graficación instantánea de ellos, en cambio utilizando el Traker tienen una opción llamada Auto-Track, que realiza automáticamente la recolección de datos pero depende de la calidad del video, que se debe grabar previamente. Por otro lado, en el software libre Python se pueden incorporar una serie de librerías que permiten mediante la elaboración de scripts, el ajuste matemático y simulación de los datos experimentales.

Ahora bien, al comparar los resultados obtenidos con la calculadora científica Casio ClassPad 300, y los software Tracker y Python, se comprobó que la selección del analizador de video Tracker, además del software Python, pueden ser buenas herramientas didácticas para hacer experimentos y simulaciones de este fenómeno físico como es la caída libre de los cuerpos, con bajos recursos económicos.

V. REFERENCIAS

[1] Sánchez, M., *Siete cuestiones para divulgar y comprender aspectos de la caída libre*, *Latin American Journal of Physics Education* **5**, 1 – 10 (2011).
 [2] Ríos, V., *Utilización del software Modellus en la Física del Programa Nacional de Formación de Ingeniería en Materiales Industriales del Instituto Universitario de Tecnología de Maracaibo*, Trabajo de Ascenso, Instituto Universitario de Tecnología de Maracaibo, Maracaibo, Venezuela, (2015).
 [3] Begeler, E., *Experiment showing the motion of a falling object and the influence of air drag*. *Latin American Journal of Physics Education* **3**, 52-54 (2009).

Montero Germán

[4] Montero G., *Equivalente electromecánico del calor*. *Latin American Journal of Physics Education* **3**, 299-303 (2014).

[5] Serway, R., Jewett, J., *Física para Ciencias e Ingeniería*, 7ª Edición. (Editorial Cengage-Learning, México, 2009).

[6] Tipler, P., Mosca, G., *Física para la Ciencia y la Tecnología*, 7ª Edición, (Reverte S.A, España, 2005).

[7] Marion, J., *Dinámica Clásica de las Partículas y Sistemas*, 2ª Edición (Editorial Reverte, España, 1975), p. 653.

[8] Originlab, *Data Analysis and Graphing Software*, www.originlab.com.

[9] MatLab. <http://www.mathworks.com>.

[10] Mathematica. www.wolfram.com/mathematica.

[11] Ohanian, H., Market, J., *Física para Ciencias e Ingeniería*, 3ª Edición, (Editorial McGraw Hill / Interamericana, México, 2009).