



**Universidad de Valladolid**



**ESCUELA DE INGENIERÍAS  
INDUSTRIALES**

**UNIVERSIDAD DE VALLADOLID  
ESCUELA DE INGENIERIAS INDUSTRIALES**

**GRADO EN INGENIERÍA EN TECNOLOGÍAS INDUSTRIALES**

**ANÁLISIS DE EXPERIENCIAS FÍSICAS EN  
UNA MESA NEUMÁTICA MEDIANTE VISIÓN  
ARTIFICIAL**

**Autor:**

**Rodríguez Ares, Estela**

**Tutor:**

**Paramo Vela, José Ricardo**

**Departamento de Física  
Aplicada**

**Valladolid, julio de 2024**



## RESUMEN Y PALABRAS CLAVE

En este trabajo se ha desarrollado un programa capaz de detectar las posiciones que toma un volante en una mesa neumática a lo largo del tiempo, mediante el uso del software MATLAB. El programa tiene como objetivo abordar una limitación surgida en la realización de prácticas de laboratorio de los estudiantes de primer curso, debido a la interrupción en la fabricación de un componente clave. Gracias a la visión artificial, se puede reanudar su uso en los estudios cinemáticos y dinámicos.

### Palabras clave:

Visión Artificial, Detección de Objetos, Seguimiento de Objetos, Prácticas de Laboratorio, MATLAB



## ABSTRACT & KEYWORDS

In this work, a program has been developed to detect the positions of a flywheel on a pneumatic table over time using MATLAB software. This program aims to address a limitation that arose during the laboratory practices of first-year students due to the discontinuation of a key component. Through artificial vision, its use in kinematic and dynamic studies can be resumed.

### Keywords:

Artificial Vision, Object Detection, Object Tracking, Laboratory Practices, MATLAB



# ÍNDICE

<b>CAPÍTULO 1. INTRODUCCIÓN</b> .....	1
1.1. CONTEXTO.....	1
1.2. PLANTEAMIENTO DEL PROBLEMA Y OBJETIVOS .....	2
1.3. ESTRUCTURA DEL TRABAJO .....	4
<b>CAPÍTULO 2. ESTADO DEL ARTE</b> .....	5
VISIÓN ARTIFICIAL .....	5
PROCESAMIENTO DE IMÁGENES .....	5
DETECCIÓN DE OBJETOS .....	6
SEGUIMIENTO DE OBJETOS .....	6
<b>CAPÍTULO 3. INSTRUMENTACIÓN Y MODO DE OPERAR</b> .....	7
3.1. INSTRUMENTACIÓN .....	7
3.2. MODO DE OPERAR .....	10
<b>CAPÍTULO 4. PROGRAMACIÓN EN MATLAB</b> .....	11
4.1. FUNCIÓN EXTERNA → DETECCIÓN DEL VOLANTE.....	11
4.2. FUNCIÓN PRINCIPAL → SEGUIMIENTO DEL MOVIMIENTO .....	16
CÁLCULO DE LA POSICIÓN .....	18
CÁLCULO DE LA VELOCIDAD .....	24
CÁLCULO DE LA ACELERACIÓN .....	26
<b>CAPÍTULO 5. ESTUDIO DE MOVIMIENTOS</b> .....	29
5.1. CONCEPTOS BÁSICOS CINEMÁTICA.....	29
5.2. MOVIMIENTO RECTILÍNEO .....	30
<i>MARCO TEÓRICO</i> .....	30
<i>EXPLICACIÓN DEL M.R. REALIZADO</i> .....	30
RESULTADO TEÓRICO.....	31
<i>RESULTADO EXPERIMENTAL PAPEL ELECTROCONDUCTOR</i> .....	40
5.3. MOVIMIENTO RECTILÍNEO UNIFORMEMENTE VARIADO .....	47
<i>MARCO TEÓRICO</i> .....	47
<i>EXPLICACIÓN DEL M.R.U.V. REALIZADO</i> .....	48
RESULTADO TEÓRICO.....	50
<i>RESULTADO EXPERIMENTAL MATLAB</i> .....	52
<i>RESULTADO EXPERIMENTAL PAPEL ELECTROCONDUCTOR</i> .....	59

COMPARACIÓN.....	63
<b>CAPÍTULO 6. CONCLUSIONES .....</b>	<b>65</b>
6.1. CONCLUSIONES GENERALES .....	65
6.2. LÍNEAS FUTURAS .....	66
<b>BIBLIOGRAFÍA .....</b>	<b>67</b>

## ANEXOS

ANEXO I. FUNCIÓN PRINCIPAL. SEGUIMIENTO DEL MOVIMIENTO.

ANEXO II. FUNCIÓN EXTERNA. DETECCIÓN DEL VOLANTE.

## LISTADO DE TABLAS

Tabla 1. Posición del volante en el tiempo. MR. MATLAB. ....	33
Tabla 2. Velocidad del volante en el tiempo. MR. MATLAB. ....	35
Tabla 3. Aceleración del volante en el tiempo. MR. MATLAB. ....	38
Tabla 4. Datos del volante en el tiempo. MR. Papel electroconductor.....	41
Tabla 5. Posición del volante en el tiempo. MRUV. MATLAB.....	52
Tabla 6. Velocidad del volante en el tiempo. MRUV. MATLAB.....	54
Tabla 7. Aceleración del volante en el tiempo. MRUV. MATLAB. ....	57
Tabla 8. Datos del volante en el tiempo. MRUV. Papel electroconductor.....	60

## LISTADO DE FIGURAS

Figura 1. Mesa neumática.....	2
Figura 2. Descripción mesa neumática.....	8
Figura 3. Fragmento de código: Entradas.....	11
Figura 4. Representación capas imagen RGB.....	12
Figura 5. Fragmento de código: RGB.....	12
Figura 6. Fragmento de código: Detección del volante.....	13
Figura 7. Ejemplo detección del volante.....	13
Figura 8. Fragmento de código: Cierre morfológico.....	14
Figura 9. Ejemplo cierre morfológico.....	14
Figura 10. Fragmento de código: Eliminación de ruido.....	14
Figura 11. Ejemplo eliminación de ruido.....	15
Figura 12. Fragmento de código: Propiedades del volante.....	15
Figura 13. Velocidad de fotograma.....	16
Figura 14. Fragmento de código: Datos de entrada.....	17
Figura 15. Datos de entrada en la ventana de comandos.....	17
Figura 16. Fragmento de código: Fotogramas y tiempo real entre ellos.....	18
Figura 17. Fragmento de código: Bucle llamada a función externa.....	18
Figura 18. Datos de entrada para conocer las unidades de medida.....	19
Figura 19. Fragmento de código: Unidades de medida.....	20
Figura 20. Fragmento de código: Equivalencia píxeles y centímetros.....	20
Figura 21. Fragmento de código: Representación datos posiciones.....	21
Figura 22. Formato tabla de posiciones.....	21
Figura 23. Variación decimales Excel 1.....	22
Figura 24. Variación decimales Excel 2.....	22
Figura 25. Fragmento de código: Velocidad inicial.....	24
Figura 26. Fragmento de código: Bucle velocidades.....	24
Figura 27. Fragmento de código: Representación datos velocidades.....	25
Figura 28. Formato tabla de velocidades.....	25
Figura 29. Fragmento de código: Aceleración inicial.....	26
Figura 30. Fragmento de código: Bucle aceleraciones.....	26
Figura 31. Fragmento de código: Representación datos aceleraciones.....	27
Figura 32. Formato tabla de aceleraciones.....	27
Figura 33. Fragmento de código: Representación datos conjuntos.....	28
Figura 34. Formato tabla conjunta.....	28
Figura 35. Conceptos cinemática.....	29
Figura 36. Práctica MR.....	30
Figura 37. Video MR.....	31
Figura 38. Gráfica de posición. MR. MATLAB.....	33
Figura 39. Ajuste por mínimos cuadrados. Posición. MR. MATLAB.....	34
Figura 40. Velocidad frente al tiempo. MR. MATLAB.....	36
Figura 41. Ajuste por mínimos cuadrados. Velocidad. MR. MATLAB.....	36

Figura 42. Aceleración frente al tiempo. MR. MATLAB. ....	38
Figura 43. Aceleración frente al tiempo y cte. MR. MATLAB. ....	39
Figura 44. Papel electroconductor. MR. ....	40
Figura 45. Ajustes mínimos cuadrados. Posición frente al tiempo. MR. Papel electroconductor. ....	42
Figura 46. Velocidad frente al tiempo. MR. Papel electroconductor. ....	42
Figura 47. Ajustes mínimos cuadrados. Velocidad frente al tiempo. Papel electroconductor. ....	43
Figura 48. Aceleración frente al tiempo. MR. Papel electroconductor. ....	43
Figura 49. Aceleración frente al tiempo y cte. MR. Papel electroconductor. ...	44
Figura 50. Gráfica $x - t$ en M.R.U.V. [3].....	47
Figura 51. Gráfica $v - t$ en M.R.U.V. [3].....	48
Figura 52. Práctica MRUV. ....	48
Figura 53. Práctica MRUV. ....	49
Figura 54. Video MRUV. ....	49
Figura 55. Diagrama de fuerzas. MRUV.....	50
Figura 56. Gráfica de posición. MRUV. MATLAB. ....	53
Figura 57. Ajuste por mínimos cuadrados. Posición. MRUV. MATLAB.....	53
Figura 58. Velocidad frente al tiempo. MRUV. MATLAB.....	55
Figura 59. Ajuste por mínimos cuadrados. Velocidad. MRUV. MATLAB. ....	55
Figura 60. Aceleración frente al tiempo. MRUV. MATLAB. ....	57
Figura 61. Aceleración frente al tiempo y cte. MRUV. MATLAB.....	58
Figura 62. Papel electroconductor. MRUV.....	59
Figura 63. Ajustes mínimos cuadrados. Posición frente al tiempo. MRUV. Papel electroconductor. ....	61
Figura 64. Ajustes mínimos cuadrados. Velocidad frente al tiempo. MRUV. Papel electroconductor. ....	61
Figura 65. Aceleración frente al tiempo y cte. MRUV. Papel electroconductor. ....	62

# CAPÍTULO 1. INTRODUCCIÓN

## 1.1. CONTEXTO

¿Alguna vez te has preguntado cómo las máquinas pueden "ver" el mundo que nos rodea? La respuesta se encuentra en la visión artificial, un campo fascinante que combina la ciencia, la tecnología y la creatividad para dotar a las máquinas de la capacidad de entender imágenes y videos, como lo hacen los seres humanos

En la actualidad, la visión artificial se ha convertido en una pieza fundamental de nuestro día a día, transformando la forma en que interactuamos con el mundo que nos rodea. Desde la detección de rostros de los móviles, hasta los vehículos autónomos capaces de percibir obstáculos en la carretera.

Esta tecnología se basa en la capacidad de capturar, analizar y extraer información del entorno mediante el procesamiento de imágenes y videos. Esto incluye la detección de objetos, la identificación de patrones, la medición de características y la interpretación de movimientos.

Uno de los conceptos clave en la visión artificial es la detección y el seguimiento de objetos, parte importante en la realización de este trabajo.

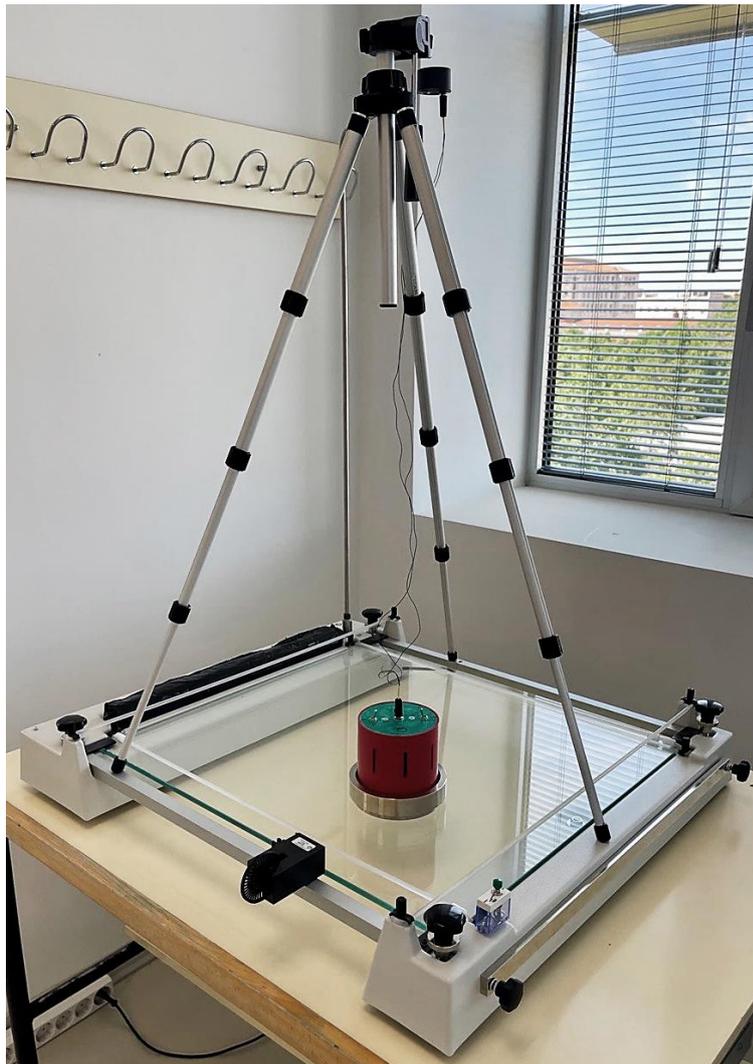
La detección de objetos es la capacidad de identificar y localizar objetos específicos en una imagen o video. Esta función es esencial en aplicaciones como la seguridad, la robótica y la vigilancia, donde es necesario identificar objetos de interés en tiempo real, como también lo es en la industria para la inspección de productos.

El seguimiento de objetos, por otro lado, implica el rastreo continuo de un objeto a lo largo de una secuencia de imágenes o video. Este proceso permite seguir objetos en movimiento, lo que es crucial en aplicaciones como la navegación autónoma de vehículos, el seguimiento de objetos en videos de vigilancia y la realidad aumentada.

La visión artificial ha experimentado un crecimiento exponencial en los últimos años convirtiéndose en un campo de investigación y aplicación de gran importancia. La capacidad de las máquinas para percibir y comprender el entorno visual ha impulsado avances significativos en una amplia variedad de industrias.

## 1.2. PLANTEAMIENTO DEL PROBLEMA Y OBJETIVOS

En el ámbito educativo, la observación y el análisis de movimientos son esenciales para la comprensión de conceptos fundamentales en física. En el laboratorio de Física Aplicada de la Escuela de Ingenierías Industriales de la Universidad de Valladolid se realizan diferentes experimentos prácticos para ayudar a los alumnos a asimilar la teoría, y comprobar la correspondencia entre los resultados teóricos y experimentales. Una de las prácticas que se realizaban era un estudio cinemático gracias a una mesa neumática.



*Figura 1. Mesa neumática*

La mesa neumática está compuesta por un vidrio sobre el que se desplazan los deslizadores neumáticos prácticamente sin fricción. La fuerza de rozamiento es mínima, ya que los deslizadores poseen ventiladores capaces de crear un colchón de aire entre ellos y el vidrio. El deslizador cuenta en la parte inferior con un electrodo central, y también con uno adicional situado en la periferia.

Para el estudio de los diferentes movimientos es necesario un papel electroconductor situado entre los deslizadores y el vidrio. En este papel el electrodo marca las posiciones que va tomando a lo largo del tiempo con una frecuencia ajustable de 10 ó 50 Hz.

El papel electroconductor se dejó de fabricar, lo que ha llevado al desuso de esta práctica al no poder contar con este componente necesario.

Este presente trabajo busca abordar esta limitación, mediante el desarrollo de un sistema de seguimiento de objetos basado en el procesamiento de imágenes y videos. Proporcionando así una solución efectiva que permita a los estudiantes poder volver a experimentar y comprender la cinemática y dinámica del sólido rígido de una manera práctica.

En resumen, este trabajo se motiva por la necesidad de recuperar una práctica educativa valiosa y por el deseo de explorar y demostrar el potencial de la visión artificial en un contexto educativo y práctico. Al lograr estos objetivos, se espera que este trabajo proporcione una herramienta útil tanto en el aula como en la investigación en visión artificial.

#### OBJETIVO GENERAL

Desarrollar un sistema de seguimiento de objetos mediante el uso de visión artificial que permita la observación y análisis de diferentes movimientos en una mesa neumática. Se va a desarrollar en el software MATLAB.

#### OBJETIVOS ESPECÍFICOS

- Realizar el estado del arte de métodos utilizados en el seguimiento de objetos.
- Describir el funcionamiento de la mesa neumática, y los instrumentos necesarios para la realización de la práctica.
- Desarrollar un algoritmo de detección de objetos que permita identificar y localizar el volante en cada fotograma de la secuencia de video.
- Obtener la gráfica de posición del volante que sustituya el uso del papel electroconductor. También las gráficas de velocidad y aceleración del volante a lo largo del tiempo.
- Conseguir que la programación sea aplicable independientemente de la calidad de la cámara.
- Conseguir que las condiciones de iluminación o la altura de grabación, afecten los menos posible a los resultados obtenidos.
- Realizar pruebas y validaciones para evaluar la precisión y la efectividad del sistema de seguimiento de objetos.

### 1.3. ESTRUCTURA DEL TRABAJO

A continuación, se detalla la estructura de las secciones que lo componen, para proporcionar una visión general de la organización de este trabajo:

#### CAPÍTULO 1. INTRODUCCIÓN

Presentación del tema de investigación en el contexto actual. También se declara el problema por el que ha surgido junto con los objetivos buscados. Por último, se explica la estructura que va a tener el trabajo.

#### CAPÍTULO 2. ESTADO DEL ARTE

Explicación de los conceptos fundamentales relacionados con la visión artificial y el seguimiento de objetos, y posibles técnicas a utilizar.

#### CAPÍTULO 3. INSTRUMENTACIÓN Y MODO DE OPERAR

Descripción del material utilizado, y el procedimiento que se ha seguido a la hora de realizar las pruebas experimentales.

#### CAPÍTULO 4. PROGRAMACIÓN EN MATLAB

Programación realizada para la detección del volante, y el seguimiento a lo largo del tiempo de este.

#### CAPÍTULO 5. ESTUDIO DE MOVIMIENTOS

Conceptos fundamentales de la cinemática, y estudio de cada uno de los movimientos realizados. En cada uno de ellos se incluye su marco teórico, los datos teóricos, los datos experimentales obtenidos y su correspondencia entre ellos.

#### CAPÍTULO 6. CONCLUSIONES

Presentación de las conclusiones obtenidas, y de los objetivos conseguidos. También se plantearán direcciones futuras de investigación.

## CAPÍTULO 2. ESTADO DEL ARTE

### VISIÓN ARTIFICIAL

La visión artificial ha sido un área de investigación activa en la última década. A medida que la tecnología ha avanzado, se han logrado avances significativos en la detección y el seguimiento de objetos en diversos contextos. A continuación, se presenta un resumen de los principales enfoques y desarrollos relacionados con el seguimiento de objetos, con un énfasis en las áreas pertinentes para el presente trabajo.

La visión artificial es una disciplina interdisciplinaria que combina la informática, las matemáticas y la ingeniería que permite a las máquinas interpretar y comprender imágenes y videos. Algunos de los conceptos fundamentales de la visión artificial incluyen:

- **Procesamiento de Imágenes:** El procesamiento de imágenes implica la manipulación de imágenes digitales para realzar sus características, eliminar el ruido y extraer información útil. Esto puede incluir operaciones como la filtración, la segmentación y la transformación de imágenes.
- **Detección de Objetos:** La detección de objetos es la capacidad de identificar y localizar objetos específicos en una imagen o secuencia de video. Los algoritmos de detección de objetos se basan en características visuales, como bordes, colores y texturas, para realizar esta tarea.
- **Seguimiento de Objetos:** El seguimiento de objetos implica rastrear un objeto a medida que se mueve a lo largo de una secuencia de imágenes o video. Esto es esencial para seguir el movimiento de objetos en tiempo real

### PROCESAMIENTO DE IMÁGENES

- **Preprocesamiento de imágenes:** Antes de que las imágenes sean útiles para el análisis, a menudo se aplican técnicas de preprocesamiento, como corrección de color, eliminación de ruido y ajuste de contraste.
- **Segmentación de imágenes:** La segmentación se refiere a la separación de una imagen en regiones de interés, como objetos y fondos.
- **Métodos de filtrado:** Para mejorar la precisión del seguimiento de objetos, es común aplicar métodos de filtrado, como el filtro de Kalman

o el filtro de partículas. Estos métodos estiman la posición futura del objeto y corrigen las predicciones a medida que se obtiene nueva información.

## DETECCIÓN DE OBJETOS

La detección de objetos es el proceso de identificar y localizar objetos en una imagen. Algunas técnicas comunes de detección de objetos incluyen:

- **Detección de bordes:** Identifica cambios abruptos en la intensidad de los píxeles, lo que puede indicar la presencia de un objeto.
- **Detección de color:** Busca regiones con colores específicos que coincidan con el objeto de interés.
- **Detección basada en características:** Utiliza descriptores de características como SIFT o SURF para encontrar objetos en imágenes.
- **Detección basada en aprendizaje profundo:** Emplea redes neuronales convolucionales (CNN) para aprender automáticamente las características relevantes de los objetos.

## SEGUIMIENTO DE OBJETOS

El seguimiento de objetos implica el rastreo continuo de un objeto en movimiento a lo largo de una secuencia de imágenes. Algunos enfoques comunes para el seguimiento de objetos incluyen:

- **Seguimiento de puntos clave:** Identifica puntos clave en el objeto y realiza un seguimiento de su posición a lo largo del tiempo.
- **Seguimiento óptico:** Utiliza las características del objeto y la información de movimiento para predecir la posición del objeto en el siguiente fotograma.
- **Filtros de seguimiento:** Pueden utilizarse filtros de Kalman o filtros de partículas para mejorar la precisión del seguimiento.

## CAPÍTULO 3. INSTRUMENTACIÓN Y MODO DE OPERAR

### 3.1. INSTRUMENTACIÓN

#### COMPONENTES DE UN SISTEMA DE VISIÓN ARTIFICIAL

- SISTEMA DE ADQUISICIÓN DE IMAGEN

El sistema de adquisición de imagen es el primer paso en cualquier sistema de visión artificial y se encarga de capturar imágenes del entorno o de los objetos de interés que serán procesadas y analizadas posteriormente. Este sistema incluye varios componentes esenciales: cámaras, lentes, iluminación y sensores de imagen.

En concreto, en este proyecto se ha utilizado una cámara de móvil que capta a 30 FPS (fotogramas por segundo). Y se ha programado para poderse adaptar a cualquier cámara.

Se han realizado grabaciones en distintos horarios para asegurarse que la iluminación natural no interfería en unos resultados óptimos.

- SISTEMA DE PROCESAMIENTO

El sistema de procesamiento se encarga de analizar las imágenes capturadas para extraer información relevante. En este proyecto se ha utilizado el software Matlab para realizar el seguimiento del objeto.

- SISTEMA MECÁNICO

El sistema mecánico es responsable de soportar y posicionar los componentes del sistema de visión artificial. Este sistema asegura que las imágenes se capturen de manera precisa y repetible.

Se ha utilizado un trípode capaz de sostener la cámara en una posición fija durante la grabación del experimento.

Los elementos necesarios para la realización de las pruebas son los siguientes:

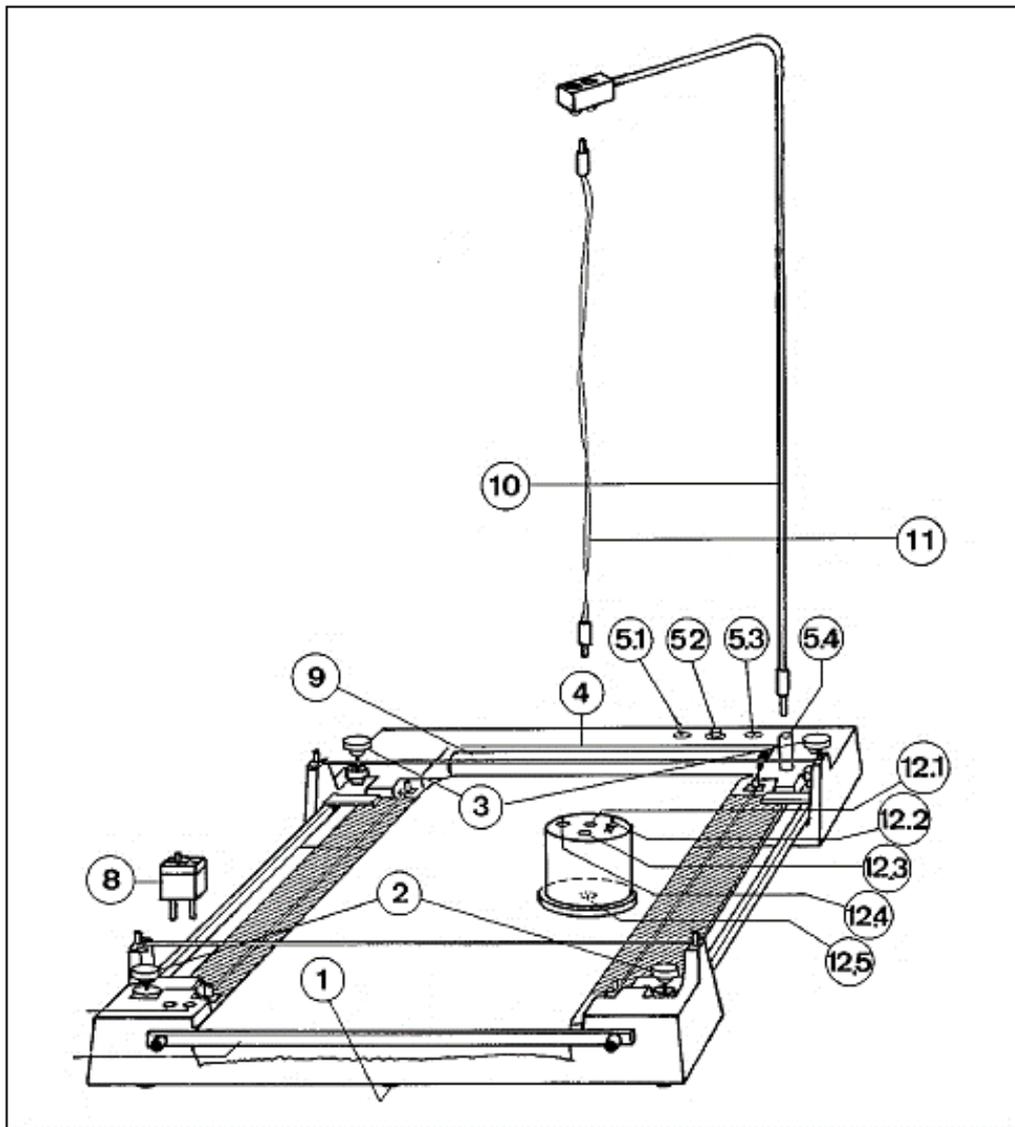


Figura 2. Descripción mesa neumática

- (1) Estructura de la mesa
- (2,3) Tornillos de ajuste niveladores
- (4) Hueco para el papel metalizado
- (5) Alimentación 220v/50 Hz
  - (5.1) Interruptor de frecuencia (10/50 Hz)
  - (5.2) Interruptor de alimentación
  - (5.3) Fusible primario
  - (5.4) Interruptor de alimentación
- (8) Interruptor para proporcionar los impulsos eléctricos
- (9) Papel electroconductor

- (10) Brazo de alimentación enchufable en 5.4
- (11) Cable de alimentación de 85 cm de longitud
- (12) Deslizador con ventilador y electrodo central
  - (12.1) Enchufe para cable 11
  - (12.2) Interruptor ventilador ON/OFF
  - (12.3) Toma para conectar el electrodo adicional
  - (12.4) Interruptor impulsos electrodo central ON/OFF
  - (12.5) Electrodo central

#### *MESA NEUMÁTICA*

Estructura exterior nivelable de dimensiones 79 x 67 x 130 cm, peso aproximado 18 kg. Cuenta con una placa de vidrio (de dimensiones 60 x 55 cm) como superficie de experimentación bordeada por bandas elásticas para limitarla.

#### *DESLIZADOR (VOLANTE)*

Cilindro deslizante (12) capaz de simular desplazamientos con fuerza de rozamiento prácticamente nula. Debido a unos ventiladores incorporados en la zona inferior que crean un colchón de aire entre ambos.

También cuenta con un electrodo (12.5) capaz de registrar las trayectorias que sigue. Esto es posible porque el electrodo realiza una descarga de corriente a intervalos fijos de tiempo sobre el papel electroconductor situado sobre el cristal.

La frecuencia de estas descargas se puede elegir dos valores fijos (interruptor 5.1), puede realizarse a 10/50 Hz.

Diámetro; 10 cm  
Altura: 10 cm  
Masa: 940 g + 490 g

### 3.2. MODO DE OPERAR

En primer lugar, se comprueba la horizontalidad de la mesa neumática y se ajusta con los tornillos niveladores para conseguir que los experimentos tengan la mayor precisión posible. Cualquier desnivel haría aumentar o disminuir la velocidad del deslizador.

Se coloca el trípode junto con el móvil para poder captar los movimientos del deslizador, y posteriormente poder realizar el procesamiento de ese video.

Con todo debidamente conectado se acciona el interruptor 12.2 del deslizador que creará el colchón de aire.

Se realizan y graban los movimientos. Por último, se introducen estas grabaciones en el programa de Matlab diseñado para poder obtener las gráficas de las posiciones tomadas por el deslizador a lo largo del tiempo en cada uno de los experimentos.

En caso de que se quisiera realizar el experimento original con el papel electroconductor, para contrastar los resultados, habría que situarlo entre el deslizador y el vidrio. Y accionar el interruptor 12.4 junto con el 12.2 para que se produzcan las descargas desde el electrodo central. Se realizará el movimiento, y en los periodos que se quiera registrar, se presionará el interruptor 8 para proporcionar los impulsos eléctricos.

Durante la realización de este trabajo se realizaron los dos experimentos, con el programa de Matlab y el papel electroconductor a la vez, para comprobar la correspondencia de los datos obtenidos.

## CAPÍTULO 4. PROGRAMACIÓN EN MATLAB

La programación realizada consta de una función principal, en el que se introducen datos de entrada, y una función externa que es llamada por la principal y se encarga de hallar la posición del deslizador en cada fotograma.

### EJECUCIÓN DEL PROGRAMA

Para obtener los resultados en MATLAB es necesario que se encuentren en la misma carpeta los archivos de la función principal “FuncionPrincipal SeguimientoDelMovimiento.m”, y la externa “FuncionExterna\_Detección DelVolante.m”, junto con el Excel ‘TablaDatosExperimentales.xlsx’ y el video del movimiento a analizar. Al ejecutar la función principal en Matlab, se pedirán unos argumentos de entrada en la ventana de comandos, que permitirán obtener los resultados deseados.

#### 4.1. FUNCIÓN EXTERNA → DETECCIÓN DEL VOLANTE

La función tiene como objetivo hallar la posición del volante y el área de la parte superior en un determinado fotograma del video. La posición es necesaria para obtener la gráfica del movimiento realizado. Matlab obtiene las posiciones y velocidades en pixeles por lo que es necesaria el área media para hallar la relación entre pixeles y medidas reales.

Esta función externa tiene como argumento de entrada una imagen, y como argumentos de salida el área y el centroide del deslizador.

```
function[centroides,area] = Posicion_imagen(im_entrada)
```

*Figura 3. Fragmento de código: Entradas*

En primer lugar, para poder realizar un correcto seguimiento del volante es necesario realizar la segmentación del fotograma, aislando el volante del fondo con el objetivo de poder determinar su posición.

La imagen introducida en la función ‘im\_entrada’ es en color, RGB. Una imagen en RGB es una composición de tres colores, para poder seleccionar una tonalidad en concreto, se guarda en tres parámetros diferentes una representación de cada color (rojo, verde y azul) que forman la imagen, como se puede comprobar en la figura 4 tomada como ejemplo.

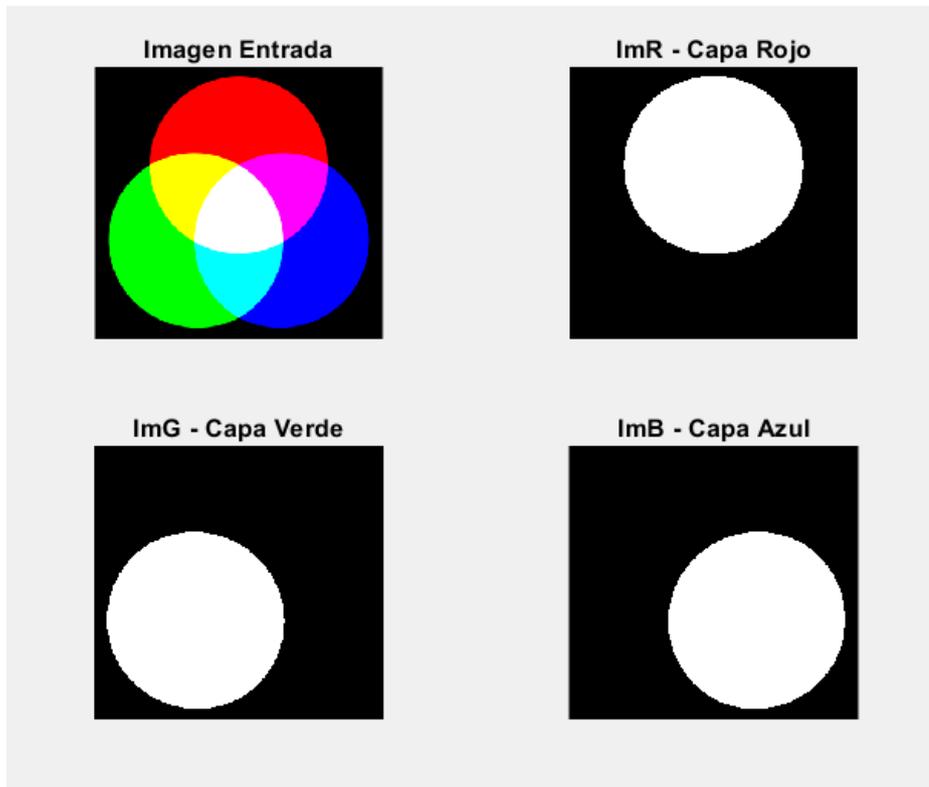


Figura 4. Representación capas imagen RGB

La separación de las capas es necesaria para la detección del volante por su color, para ello se crean umbrales en rojo, verde y azul.

También se guarda en N la altura y en M la anchura en pixeles de la imagen.

```

imR = double(im_entrada(:,:,1));
imG = double(im_entrada(:,:,2));
imB = double(im_entrada(:,:,3));

[N,M,Z] = size(im_entrada);
Y = ones(N,M);

Tr1 = 0;
Tr2 = 60;
Tg1 = 70;
Tg2 = 130;
Tb1 = 50;
Tb2 = 110;

```

Figura 5. Fragmento de código: RGB

Utilizando un bucle for se recorre todos los pixeles de la imagen, comprobando si se encuentra ese píxel entre los umbrales de los tres colores, de ser así se

igual a ese píxel a 1 haciendo que sea blanco. Si por el contrario no se encuentra entre los umbrales deseados se iguala a 0 haciendo que ese píxel sea negro.

```

for i=1:N;
    for j=1:M;
        if ((imR(i,j) > Tr1 && imR(i,j) < Tr2) && (imG(i,j) > Tg1 && ...
            imG(i,j) < Tg2) && (imB(i,j) > Tb1 && imB(i,j) < Tb2))
            Y(i,j) = 1;
        else
            Y(i,j) = 0;
        end
    end
end
end

```

Figura 6. Fragmento de código: Detección del volante

En la figura 7 se muestra como quedaría el fotograma después de la segmentación. Se puede comprobar que no es una detección perfecta del volante, hay zonas que no se ajustan a los umbrales. Se ha probado a ampliar los umbrales para reconocer estas áreas, dando como consecuencia la detección de otras zonas del fondo que no pertenecen al volante en posteriores grabaciones con diferente iluminación.

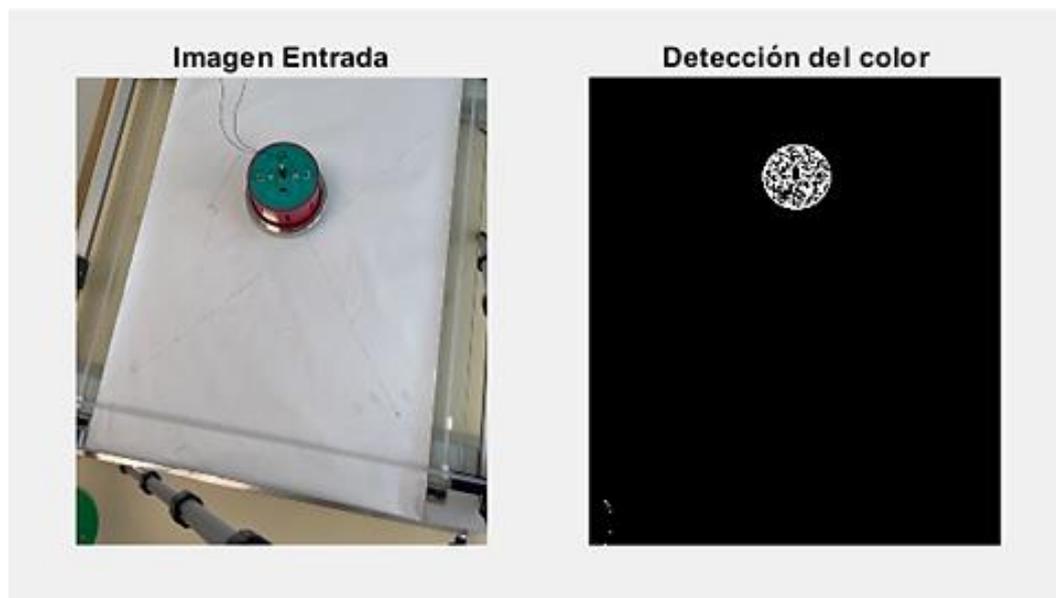


Figura 7. Ejemplo detección del volante

Por lo tanto, se han fijado los umbrales adecuados, y realizado un preprocesamiento del fotograma para la definición exacta del volante, y la reducción en el caso de existir algún ruido en el fondo de la imagen.

La definición exacta del volante se realiza rellenando las zonas no detectadas que sí pertenecen al volante.

Se configura la estructura de un disco, 'se', especificando un radio de pixeles para rellenar. A continuación, se realiza la operación cierre morfológico sobre la imagen, que consiste en una erosión precedida de una dilatación, utilizando la estructura 'se'.

```
se = strel('disk',10);  
imagen_cerrada = imclose(Y,se);
```

Figura 8. Fragmento de código: Cierre morfológico

En la figura 9 se puede comprobar cómo se ha perfeccionado el espacio ocupado por el volante.



Figura 9. Ejemplo cierre morfológico

Como se puede comprobar en la parte inferior izquierda de la figura 9 hay que realizar una eliminación del ruido. Existen partes detectadas que pertenecen al fondo, y pueden generar datos erróneos al hallar la posición.

La eliminación de ruido se realiza con la función de Matlab 'bwareaopen' sustrayendo los componentes de la imagen que contengan menos de un determinado número de pixeles.

Al realizar pruebas con diferentes calidades de cámara, y alturas del trípode, se ha determinado que el área que ocupan los pixeles del volante suele estar entre 5000 y 7000. Por lo tanto, se ha obtenido un resultado satisfactorio al eliminar de la imagen toda detección que tenga un tamaño inferior de 3000 pixeles.

```
bw = bwareaopen(imagen_cerrada,3000);
```

Figura 10. Fragmento de código: Eliminación de ruido

En la figura 11 se puede ver como finalmente se elimina el ruido de la imagen y el proceso que ha seguido el programa para el preprocesamiento del fotograma para obtener los resultados deseados en la detección del volante.

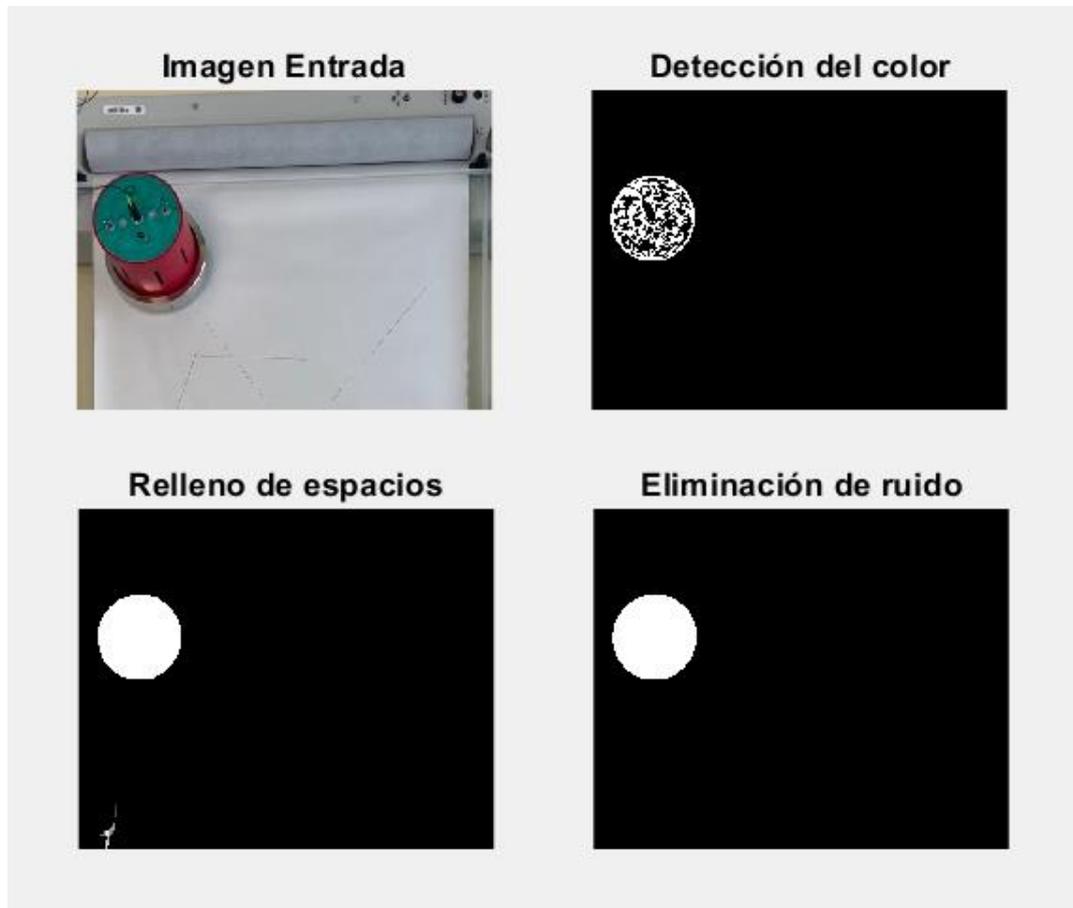


Figura 11. Ejemplo eliminación de ruido

Una vez realizada la detección del volante el necesario conocer sus propiedades. Se determina el centro de masas de la imagen binaria, que determina la posición del centro del volante en cada fotograma introducido en la función. Y también el área del volante en cada momento. Ambos son los argumentos de salida de la función externa que se recibirán al llamarla desde la función principal.

```
propiedades = regionprops(bw);
centroide = propiedades.Centroid;
area=propiedades.Area;
```

Figura 12. Fragmento de código: Propiedades del volante

## 4.2. FUNCIÓN PRINCIPAL → SEGUIMIENTO DEL MOVIMIENTO

La función principal tiene como objetivo realizar el seguimiento de las posiciones que va tomando el volante a lo largo del tiempo. La programación realizada se puede adaptar a diferentes calidades de video, o frecuencias de la gráfica de posiciones deseada. Con este objetivo en el inicio del programa se piden ciertos datos de entrada:

- La ubicación donde se encuentra el video en el ordenador.
- Los fotogramas por segundo que aparecen en las propiedades del video. Se puede comprobar al pulsar clic derecho en el video, y nuevamente en propiedades.

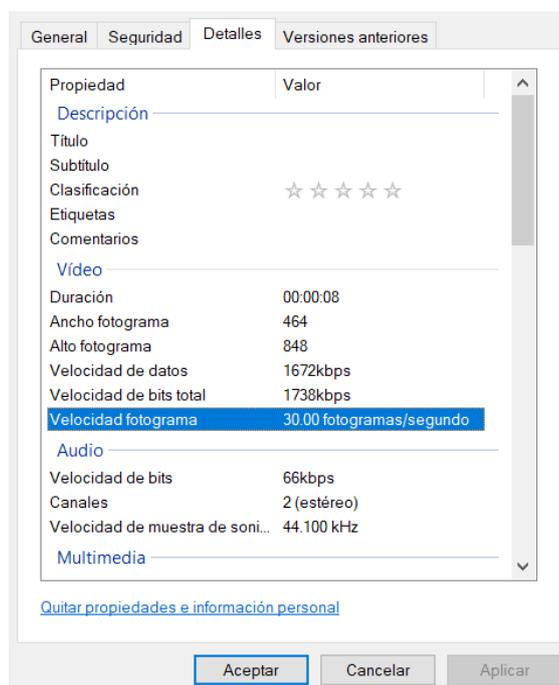


Figura 13. Velocidad de fotograma

- El espaciado entre fotogramas, en función de la frecuencia con la que se quieran conocer las posiciones del volante. Si se tomara la posición de cada fotograma los puntos de posición estarían demasiado cerca. Este espaciado deberá variar en función de la velocidad del volante hasta obtener los resultados deseados.

A la hora de garantizar que el formato de los datos introducidos sea adecuado para el funcionamiento del programa, los datos de entrada son pedidos junto con un ejemplo de su correcta escritura.

```

%% _1. DATOS DE ENTRADA_
% Añadir la ubicación del video:
ubi = "Ubicación del archivo con su extension \nEjemplo: \nD:\\\" + ...
      "Escritorio\\TFG VISION ARTIFICIAL\\MR.mp4\n";
txt = input(ubi,"s");
video_entrada = VideoReader(txt);

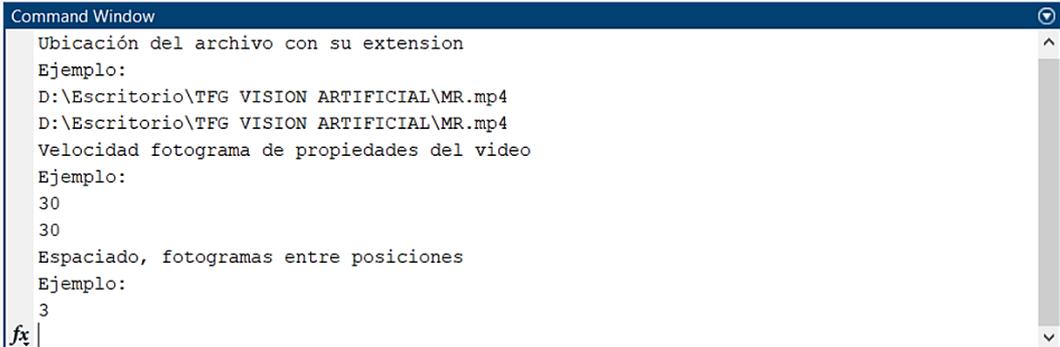
% Añadir el número de fotogramas por segundo que aparece en los detalles de
% las propiedades del video.
pedir = "Velocidad fotograma de propiedades del video \nEjemplo: \n30\n";
fotosexseg = input(pedir);

% Añadir la frecuencia deseada entre los puntos de la gráfica de posición.
pedir = "Espaciado, fotogramas entre posiciones \nEjemplo: \n3 \n";
espaciado_fotogramas = input(pedir);

```

Figura 14. Fragmento de código: Datos de entrada

En la figura 15 se puede observar cómo se visualiza en la ventana de comandos el fragmento de código anterior.



```

Command Window
Ubicación del archivo con su extension
Ejemplo:
D:\Escritorio\TFG VISION ARTIFICIAL\MR.mp4
D:\Escritorio\TFG VISION ARTIFICIAL\MR.mp4
Velocidad fotograma de propiedades del video
Ejemplo:
30
30
Espaciado, fotogramas entre posiciones
Ejemplo:
3
fx |

```

Figura 15. Datos de entrada en la ventana de comandos.

## CÁLCULO DE LA POSICIÓN

Partiendo de estos datos de entrada se continúa con el procesamiento del video. En primer lugar, se halla el número de fotogramas que contiene para poder introducirlos en la función externa. También se establece el espaciado de tiempo real en segundos para obtener los datos del volante, en función de la calidad del video.

```
N = video_entrada.NumFrames;  
tiempo = (1/fotoxseg)*espaciado_fotogramas;
```

Figura 16. Fragmento de código: Fotogramas y tiempo real entre ellos

Se crean vectores vacíos que posteriormente alojen las áreas, el tiempo y las posiciones de los centroides del volante a lo largo del tiempo. Se inicializa el parámetro del instante del tiempo (t) por si tiene algún valor previo que no interfiera en los resultados.

Se establece un bucle for que captura la imagen de un fotograma cada intervalo de segundos establecido en los datos de entrada. Una vez obtenida esta imagen, se procesa mediante la función de posición, la cual devuelve el centroide del volante. Estos valores se van acumulando en los vectores de posiciones.

```
Cx = [];  
Cy = [];  
T = [];  
t = 0;  
area = [];  
  
for i = 1:(espaciado_fotogramas):N  
    imagen = read(video_entrada,i);  
    t = t + tiempo;  
    [C,a] = Posicion_imagen(imagen); % Llamada a la función  
    if(~isempty(C)) % Solo se almacena si ha detectado el volante  
        Cx = [Cx ; C(:,1)]; % Almacena la posición de las x  
        Cy = [Cy ; C(:,2)]; % Almacena la posición de las y  
        T = [T ; t];  
        area = [area ; a]; % Almacena todas las áreas del volante  
    end  
end
```

Figura 17. Fragmento de código: Bucle llamada a función externa

MATLAB cuando obtiene información de una imagen lo hace en píxeles. Para poder comparar estos resultados con los obtenidos gracias al papel electroconductor, es necesario transformarlos en unidades de medida.

Se le pide al usuario que introduzca el número que corresponda a las unidades de medida deseadas. Si el número introducido no corresponde a ninguna unidad de medida, se muestra por pantalla un mensaje de ERROR: Número introducido no válido, y mediante un bucle while se vuelve a pedir introducir el número, hasta que este se corresponda con una unidad.

```

Introducir el número correspondiente para las unidades de medida deseadas:
1 para metros
2 para decímetros
3 para centímetros
4 para milímetros
0

ERROR =

    'Número introducido no válido'

Introducir el número correspondiente para las unidades de medida deseadas:
1 para metros
2 para decímetros
3 para centímetros
4 para milímetros
|

```

*Figura 18. Datos de entrada para conocer las unidades de medida.*

Cuando el número sea uno de los solicitados, se guarda en un parámetro llamado radio el valor real del radio de la zona verde del volante, en las unidades especificadas, para posteriormente realizar la equivalencia.

```

bucle = 1;
while (bucle)
    pedir = "Introducir el número correspondiente para las unidades "+...
            "de medida deseadas: \n1 para metros\n2 para decímetros\n3 "+...
            "para centímetros\n4 para milímetros\n";
    uni = input(pedir);

    if(uni == 1)
        radio = 0.048; % Unidades en metros
        bucle = 0;
    elseif(uni == 2)
        radio = 0.48; % Unidades en decímetros
        bucle = 0;
    elseif(uni == 3)
        radio = 4.8; % Unidades en centímetros
        bucle = 0;
    elseif(uni == 4)
        radio = 48; % Unidades en milímetros
        bucle = 0;
    else
        ERROR = 'Número introducido no válido'
    end
end
end

```

Figura 19. Fragmento de código: Unidades de medida

La equivalencia se realiza con el área experimental del volante, obtenida por la media de todas las áreas halladas en cada fotograma y relacionándola con la real en verde, y en las unidades de medida anteriormente especificadas.

Una vez obtenida la equivalencia, se procede a transformar las posiciones en píxeles a la unidad de medida deseada.

```

area_media = sum(area)/length(area); % (Píxel^2) Área media del volante
area_real = radio*radio*pi; % (unidad^2) Area real de la zona verde
equiv = sqrt(area_real)/sqrt(area_media); % (unidad/píxel) equivalencia

X = Cx*equiv;
Y = Cy*equiv;

S = [];
S = sqrt(X.*X+Y.*Y); % Se halla el desplazamiento

```

Figura 20. Fragmento de código: Equivalencia píxeles y centímetros

Las posiciones obtenidas se representan gráficamente, y se presentan en una tabla consolidada. Posteriormente, se exportan a un archivo Excel para su descarga.

```

figure
plot(X,Y,'kx');
title('Posición Volante');
ylabel('Eje y')
xlabel('Eje x')
grid on;

ylim([min(Y) max(Y)]);
xlim([min(X) max(X)]);

TablaC = table(T,X,Y,S);
filename = 'TablaDatosExperimentales.xlsx';
writetable(TablaC,filename,'Sheet',1,'Range','D1')

```

Figura 21. Fragmento de código: Representación datos posiciones

Si este archivo no existe, se genera automáticamente un Excel llamado TablaDatosExperimentales.xlsx en la misma carpeta donde este ubicado el programa de MATLAB, y en la primera pestaña se guarda esta tabla de posiciones.

Con el objetivo de mejorar la visualización de los datos se ha creado un archivo llamado 'TablaDatosExperimentales.xlsx' con la estructura deseada para que al ejecutar el programa se sobrescriban los datos en este, manteniendo el formato. Los datos se han exportado en su totalidad, sin redondear, pero desde Excel se ha establecido un redondeo a 3 cifras decimales

T	X	Y
0,10	0,000	0,000
0,20	0,001	0,001
0,30	0,006	0,007
0,40	0,014	0,017
0,50	0,025	0,034
0,60	0,039	0,053
0,70	0,054	0,073
0,80	0,068	0,094
0,90	0,082	0,114
1,00	0,097	0,133

Figura 22. Formato tabla de posiciones

Este redondeo se puede ajustar en todo momento, debido a que en el archivo Excel se guarda el número con todas las cifras decimales.

Para poder variar los números decimales se selecciona con el botón derecho en la parte superior de la columna, después en formato de celdas, y se abre

una nueva ventana, se selecciona en la pestaña de número donde se pueden ajustar los decimales que se desean observar.

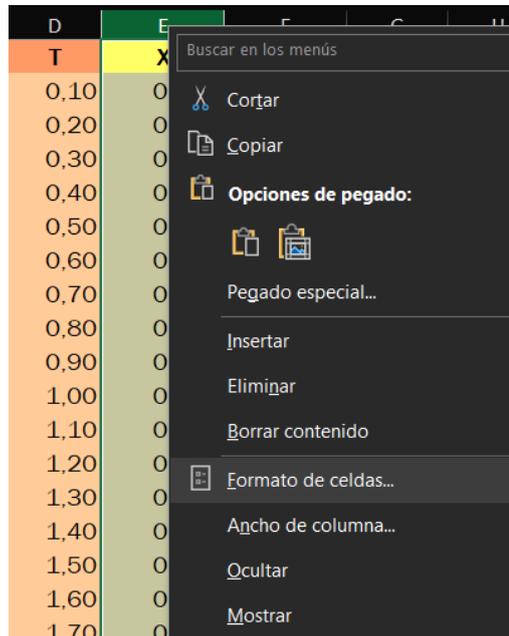


Figura 23. Variación decimales Excel 1

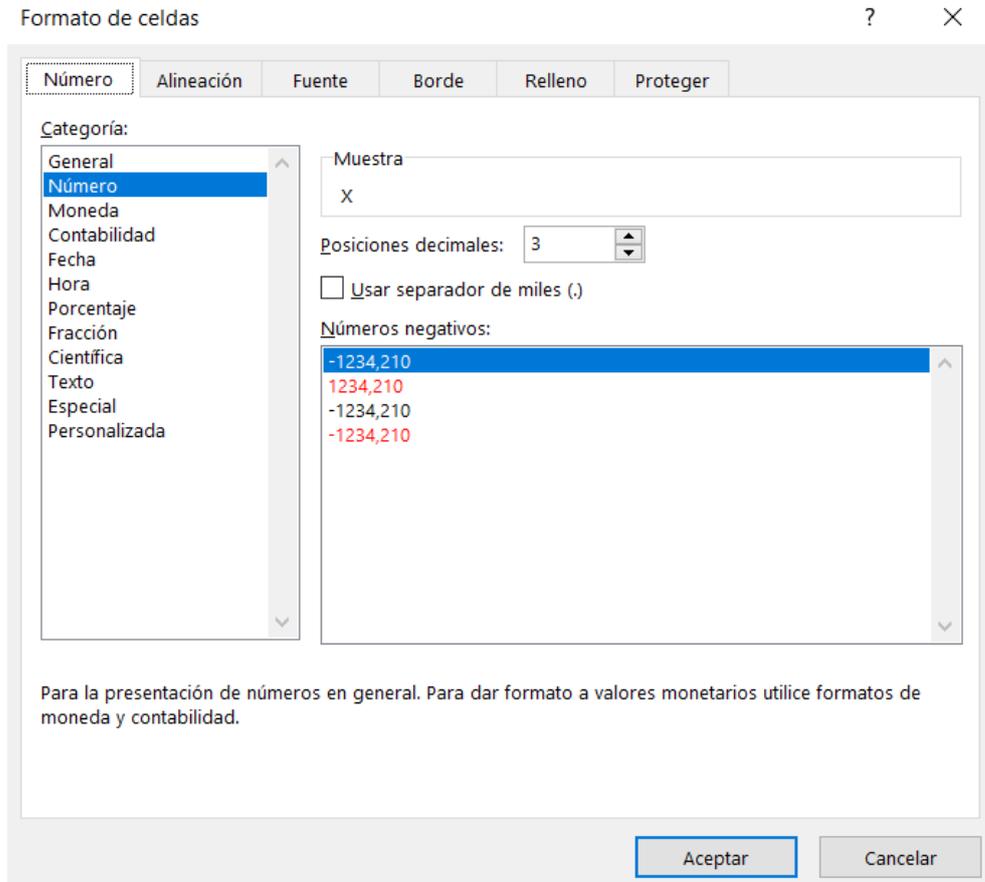


Figura 24. Variación decimales Excel 2

Es importante cerrar el archivo Excel antes de ejecutar nuevamente el programa. En caso contrario, surgirá un error y no podrá sobrescribir los datos. Si esto sucede solo habría que cerrar el archivo 'TablaDatosExperimentales.xlsx', y volver a ejecutar el programa.

## CÁLCULO DE LA VELOCIDAD

Una vez halladas las posiciones que toma el volante a lo largo del tiempo, se calculan las velocidades experimentales. En primer lugar, se establece el valor inicial de la velocidad a partir de las dos primeras posiciones y el tiempo entre ellas.

En MATLAB a la hora de trabajar con vectores es preferible que tengan el mismo número de filas. La velocidad tendría una fila menos obteniéndolo a partir de las posiciones, por lo tanto, se añade un valor más inicial.

```
vx0=(X(2)-X(1))/tiempo;  
Vx = [vx0; vx0];  
  
vy0=(Y(2)-Y(1))/tiempo;  
Vy = [vy0; vy0];  
  
v0 = sqrt(vx0*vx0+vy0*vy0);  
V = [v0; v0];
```

Figura 25. Fragmento de código: Velocidad inicial

Partiendo de estos datos se crea un bucle for que recorra los valores de los vectores de posición transformándolos en velocidades, y almacenándolos en sus respectivos vectores. La velocidad V, se obtiene al realizar la raíz cuadrada de la suma de sus componentes en el eje x, e y.

```
for i = 3:length(X)  
    m=i-1;  
    vx0=(X(i)-X(m))/tiempo;  
    Vx = [Vx ; vx0];  
    vy0=(Y(i)-Y(m))/tiempo;  
    Vy = [Vy ; vy0];  
    v0 = sqrt(vx0*vx0+vy0*vy0);  
    vn0 = (S(i)-S(m))/tiempo;  
    if (vn0<0)  
        v0 = -v0;  
    end  
    V = [V ; v0];  
end
```

Figura 26. Fragmento de código: Bucle velocidades

Se representan gráficamente las velocidades obtenidas, y se muestran en una tabla. Y al igual que las posiciones se exportan a un archivo Excel para una mejor visualización.

```
figure;
plot(T,Vx,'x');
hold on
plot(T,Vy,'x');
plot(T,V,'x');
hold off

legend('Vx','Vy','V');
title('Velocidad frente al Tiempo');
ylabel('v')
xlabel('t')
grid on;

xlim([min(T) max(T)]);

TablaV = table(T,Vx,Vy,V);
filename = 'TablaDatosExperimentales.xlsx';
writetable(TablaV,filename,'Sheet',2,'Range','D1')
```

Figura 27. Fragmento de código: Representación datos velocidades

Estos datos se almacenan en el mismo archivo anterior llamado 'TablaDatosExperimentales.xlsx', pero las velocidades en la segunda pestaña, sobrescribiendo los datos de las velocidades de otras pruebas anteriores.

	A	B	C	D	E	F	G	H	I	J
1				T	Vx	Vy	V			
2				0,10	0,014	0,013	0,019			
3				0,20	0,014	0,013	0,019			
4				0,30	0,043	0,055	0,070			
5				0,40	0,079	0,106	0,132			
6				0,50	0,113	0,162	0,198			
7				0,60	0,142	0,198	0,243			
8				0,70	0,146	0,201	0,248			
9				0,80	0,144	0,202	0,248			
10				0,90	0,143	0,199	0,245			
11				1,00	0,145	0,198	0,245			

Figura 28. Formato tabla de velocidades

Los datos de las velocidades se muestran con 3 números de decimales, pero se puede ajustar esta precisión desde Excel igual que en las posiciones.

## CÁLCULO DE LA ACELERACIÓN

Partiendo de las velocidades, se calculan las aceleraciones que toma el volante a lo largo del tiempo. En primer lugar, se establece el valor inicial de la aceleración a partir de las dos primeras velocidades y el tiempo entre ellas.

De la misma manera que ocurría con las velocidades es necesario que el vector aceleración tenga el mismo número de filas que la velocidad y la posición. Por ello, se añaden dos valores iniciales.

```
ax0=(Vx(3)-Vx(2))/tiempo;  
Ax = [ax0 ; ax0; ax0];  
  
ay0=(Vy(3)-Vy(2))/tiempo;  
Ay = [ay0 ; ay0; ay0];  
  
a0 = sqrt(ax0*ax0+ay0*ay0);  
A = [a0 ; a0; a0];
```

Figura 29. Fragmento de código: Aceleración inicial

Con estos datos se crea un bucle for que recorra los valores de los vectores de velocidad transformándolos en aceleraciones, y almacenándolos en sus respectivos vectores. La aceleración A, se obtiene al realizar la raíz cuadrada de la suma de sus componentes en el eje x, e y.

```
for i = 4:length(Vx)  
    m=i-1;  
    ax0=(Vx(i)-Vx(m))/tiempo;  
    Ax = [Ax ; ax0];  
    ay0=(Vy(i)-Vy(m))/tiempo;  
    Ay = [Ay ; ay0];  
    a0 = sqrt(ax0*ax0+ay0*ay0);  
    an0=(V(i)-V(m))/tiempo;  
    if (an0<0)  
        a0 = -a0;  
    end  
    A = [A ; a0];  
end
```

Figura 30. Fragmento de código: Bucle aceleraciones

Se representan gráficamente las aceleraciones obtenidas, y se muestran en una tabla que es exportada al mismo archivo Excel para una mejor visualización.

```
figure;
plot(T,Ax,'x');
hold on
plot(T,Ay,'x');
plot(T,A,'x');
hold off

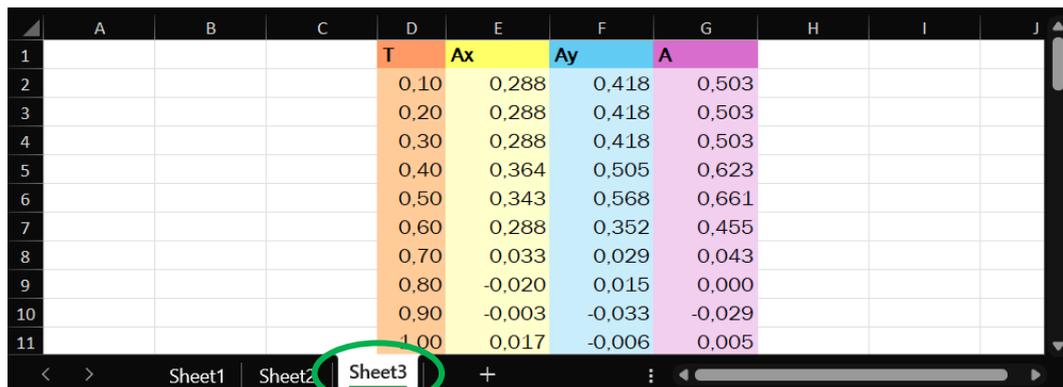
legend('Ax','Ay','A');
title('Aceleración frente al Tiempo');
ylabel('a')
xlabel('t')
grid on;

xlim([min(T) max(T)]);

TablaA = table(T,Ax,Ay,A);
filename = 'TablaDatosExperimentales.xlsx';
writetable(TablaA,filename,'Sheet',3,'Range','D1')
```

Figura 31. Fragmento de código: Representación datos aceleraciones

Estos datos se almacenan en la tercera pestaña del mismo archivo 'TablaDatosExperimentales.xlsx', sobrescribiendo los datos de las aceleraciones de otras pruebas anteriores.



	A	B	C	D	E	F	G	H	I	J
1				T	Ax	Ay	A			
2				0,10	0,288	0,418	0,503			
3				0,20	0,288	0,418	0,503			
4				0,30	0,288	0,418	0,503			
5				0,40	0,364	0,505	0,623			
6				0,50	0,343	0,568	0,661			
7				0,60	0,288	0,352	0,455			
8				0,70	0,033	0,029	0,043			
9				0,80	-0,020	0,015	0,000			
10				0,90	-0,003	-0,033	-0,029			
11				1,00	0,017	-0,006	0,005			

Figura 32. Formato tabla de aceleraciones

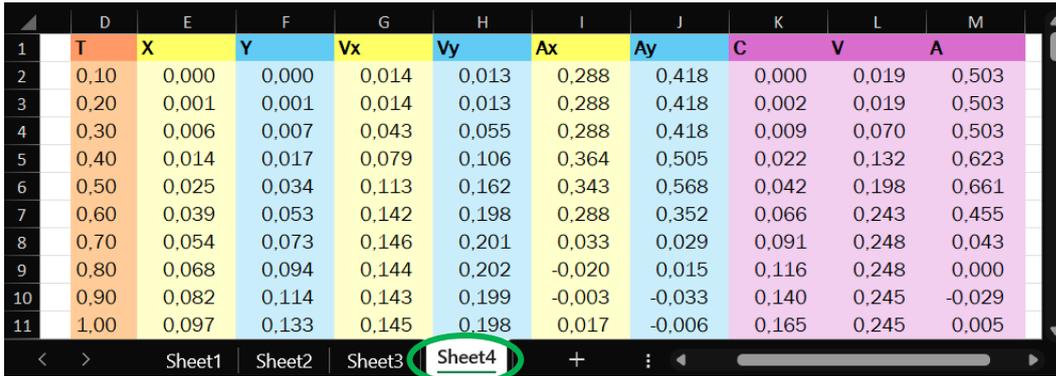
Los datos de las aceleraciones se muestran con 3 números de decimales, pero se puede ajustar esta precisión desde Excel igual que en los casos anteriores.

Por último, se crea una tabla con todos los datos obtenidos para visualizarlos conjuntamente. Esta tabla también es exportada al archivo Excel, y se

almacenan en la cuarta pestaña sobrescribiendo los datos de otras pruebas anteriores.

```
Tabla = table(T, X, Y, Vx, Vy, Ax, Ay, C, V, A)
filename = 'TablaDatosExperimentales.xlsx';
writetable(Tabla,filename,'Sheet',4,'Range','D1')
```

Figura 33. Fragmento de código: Representación datos conjuntos



	D	E	F	G	H	I	J	K	L	M
1	T	X	Y	Vx	Vy	Ax	Ay	C	V	A
2	0,10	0,000	0,000	0,014	0,013	0,288	0,418	0,000	0,019	0,503
3	0,20	0,001	0,001	0,014	0,013	0,288	0,418	0,002	0,019	0,503
4	0,30	0,006	0,007	0,043	0,055	0,288	0,418	0,009	0,070	0,503
5	0,40	0,014	0,017	0,079	0,106	0,364	0,505	0,022	0,132	0,623
6	0,50	0,025	0,034	0,113	0,162	0,343	0,568	0,042	0,198	0,661
7	0,60	0,039	0,053	0,142	0,198	0,288	0,352	0,066	0,243	0,455
8	0,70	0,054	0,073	0,146	0,201	0,033	0,029	0,091	0,248	0,043
9	0,80	0,068	0,094	0,144	0,202	-0,020	0,015	0,116	0,248	0,000
10	0,90	0,082	0,114	0,143	0,199	-0,003	-0,033	0,140	0,245	-0,029
11	1,00	0,097	0,133	0,145	0,198	0,017	-0,006	0,165	0,245	0,005

Figura 34. Formato tabla conjunta

Estos datos se muestran con 3 números de decimales, pero también se puede ajustar esta precisión desde Excel, ya que se ha exportado el número completo sin redondear.

## CAPÍTULO 5. ESTUDIO DE MOVIMIENTOS

### 5.1. CONCEPTOS BÁSICOS CINEMÁTICA

La cinemática es una rama de la física que se ocupa del estudio del movimiento de los objetos sin considerar las causas que lo producen. Se centra en describir cómo cambian la posición, velocidad y aceleración de un objeto en función del tiempo, sin entrar en detalles sobre las fuerzas involucradas.

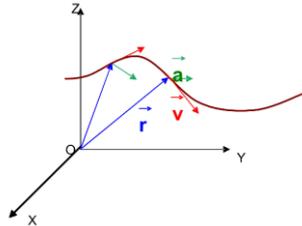


Figura 35. Conceptos cinemática

#### CONCEPTOS

**POSICIÓN:** La posición de un objeto se refiere a su ubicación en el espacio en relación con un punto de referencia. Se puede describir utilizando coordenadas cartesianas  $(x, y, z)$  en el espacio tridimensional.

**TRAYECTORIA:** Es la curva que describe la posición de un objeto en movimiento a lo largo del tiempo. Puede ser rectilínea, curva o incluso irregular.

**TIEMPO:** La variable temporal es un concepto fundamental en cinemática para estudiar cómo cambian las magnitudes físicas a lo largo del tiempo.

**DESPLAZAMIENTO:** Es el cambio neto en la posición de un objeto. Se expresa como un vector que apunta desde la posición inicial hasta la posición final. Su módulo en el Sistema Internacional (S.I.) vendrá expresado en metros (m).

**VELOCIDAD:** La velocidad es la variación de la posición de un objeto a lo largo del tiempo. Puede ser constante o variable y se expresa como la derivada de la posición con respecto al tiempo. Se utilizan habitualmente la velocidad media y la instantánea. Su unidad en el S.I. es el metro por segundo (m/s).

**ACELERACIÓN:** La aceleración es la variación de la velocidad de un objeto a lo largo del tiempo. Al igual que la velocidad, puede ser constante o variable. La aceleración positiva indica un aumento en la velocidad, mientras que la aceleración negativa indica una disminución. Se utilizan habitualmente la aceleración media y la instantánea. Se utilizan sus componentes intrínsecas, la aceleración normal y la tangencial. Su unidad en el S.I. es el metro por segundo al cuadrado ( $m/s^2$ ).

## 5.2. MOVIMIENTO RECTILÍNEO

### MARCO TEÓRICO

El movimiento rectilíneo (M.R.) es un tipo de movimiento en el que un objeto se desplaza a lo largo de una línea recta. Se busca que sea uniforme, por lo tanto, con velocidad constante y aceleración nula.

La posición de un objeto que sigue un movimiento rectilíneo uniforme cambia linealmente con el tiempo y viene dada por la siguiente ecuación:

$$s(t) = s_0 + v_0 \cdot t \quad (1)$$

$s_0 \rightarrow$  Posición inicial,  $v_0 \rightarrow$  Velocidad inicial, ambos para  $t = 0$  s.

Al representar los valores que toma la posición  $s$  frente al tiempo se obtiene una línea recta de pendiente positiva o negativa en función del signo de la velocidad.

La velocidad de un objeto que sigue un movimiento rectilíneo uniforme permanece constante  $v = v_0$ .

### EXPLICACIÓN DEL M.R. REALIZADO

Se coloca el papel electroconductor entre el volante y el cristal de la mesa neumática, a su vez se posiciona el trípode con la cámara en la parte superior para poder captar el mismo movimiento de las dos formas y así poder compararlo.

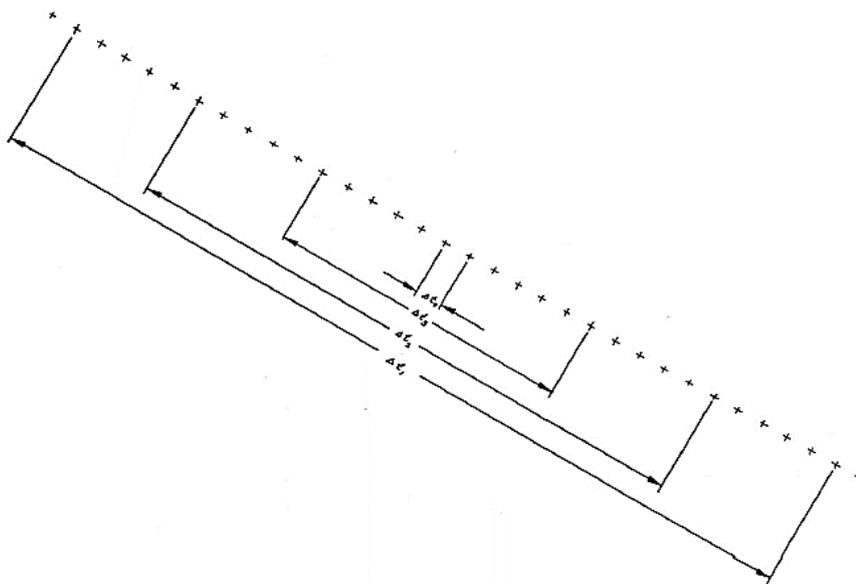


Figura 36. Práctica MR

A continuación, se enciende el interruptor y la cámara, y se realiza el lanzamiento rectilíneo del volante. En la figura 37 se puede observar el lanzamiento realizado, en la izquierda el fotograma inicial y en la derecha el fotograma final del video.

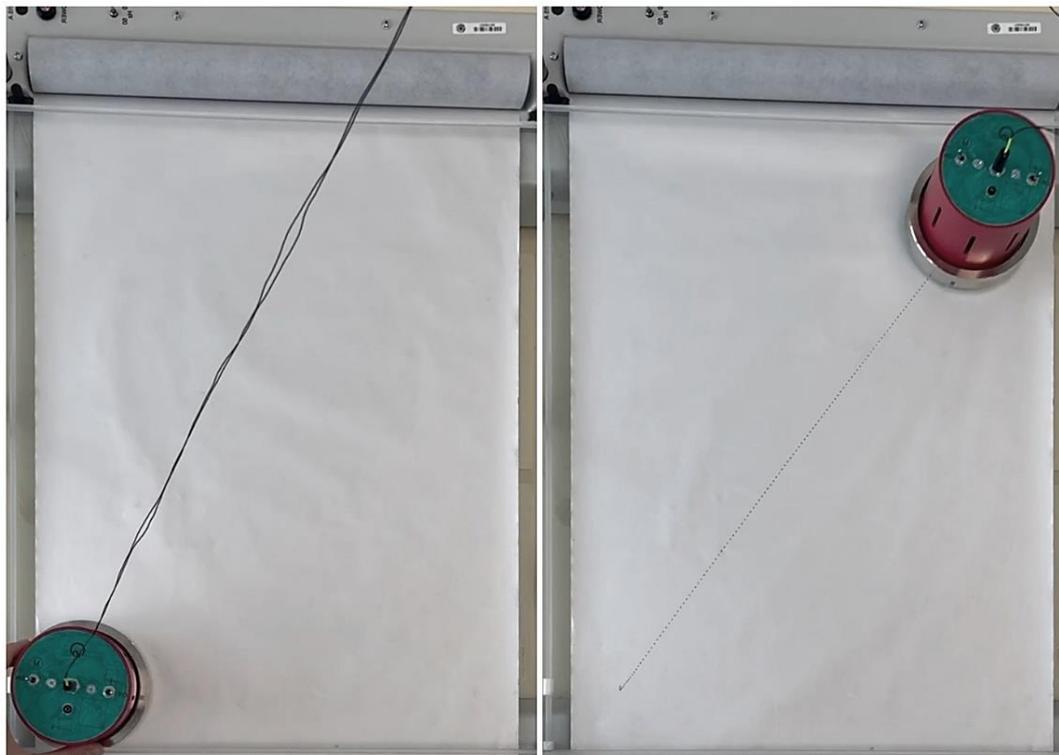


Figura 37. Video MR

A continuación, se van a representar los datos del movimiento rectilíneo obtenidos de MATLAB, a partir del video, y los del papel electroconductor, gracias al electrodo central. Y se van a comparar ambos para probar la efectividad del programa.

#### RESULTADO TEÓRICO

En este movimiento no se puede obtener un resultado teórico de las velocidades que va tomando, porque esta depende de la velocidad inicial de lanzamiento,  $V_0$ . En cada lanzamiento esa velocidad será distinta al ser impulsado inicialmente por la mano.

En un movimiento rectilíneo uniforme, en condiciones ideales, la velocidad del volante es constante en el tiempo. Por lo tanto, para intervalos de tiempos iguales debería recorrer la misma distancia.

## RESULTADO EXPERIMENTAL MATLAB

Los datos obtenidos al ejecutar el programa con el video del movimiento rectilíneo son los siguientes:

### POSICIÓN DEL VOLANTE

En la tabla 1 se muestran los desplazamientos tanto en el eje x “X (m)”, como y “Y (m)”, el módulo del desplazamiento total desde el origen de coordenadas “S (m)”.

Se puede observar como la distancia que hay entre valores de S sucesivos se reduce con el tiempo, indicando que para el mismo intervalo de tiempo cada vez recorre distancias menores, y por lo tanto el volante disminuye su velocidad con el tiempo. Por lo tanto, no se dan las condiciones ideales del movimiento rectilíneo uniforme, y existe una fuerza de rozamiento que provoca una aceleración negativa en el volante.

T ± 0,03 (s)	X ± 0,002 (m)	Y ± 0,002 (m)	S ± 0,003 (m)
0,10	0,000	0,000	0,000
0,20	0,001	0,001	0,002
0,30	0,006	0,007	0,009
0,40	0,013	0,017	0,022
0,50	0,025	0,033	0,041
0,60	0,039	0,053	0,066
0,70	0,053	0,073	0,090
0,80	0,067	0,093	0,115
0,90	0,082	0,112	0,139
1,00	0,096	0,132	0,163
1,10	0,110	0,151	0,187
1,20	0,124	0,171	0,211
1,30	0,138	0,190	0,235
1,40	0,152	0,209	0,258
1,50	0,166	0,227	0,282
1,60	0,180	0,246	0,305
1,70	0,194	0,264	0,327
1,80	0,207	0,282	0,350
1,90	0,221	0,300	0,372
2,00	0,234	0,318	0,395
2,10	0,248	0,335	0,417
2,20	0,261	0,353	0,439
2,30	0,274	0,370	0,460

2,40	0,287	0,387	0,482
2,50	0,300	0,404	0,503
2,60	0,313	0,420	0,524
2,70	0,326	0,437	0,545
2,80	0,339	0,453	0,566

Tabla 1. Posición del volante en el tiempo. MR. MATLAB.

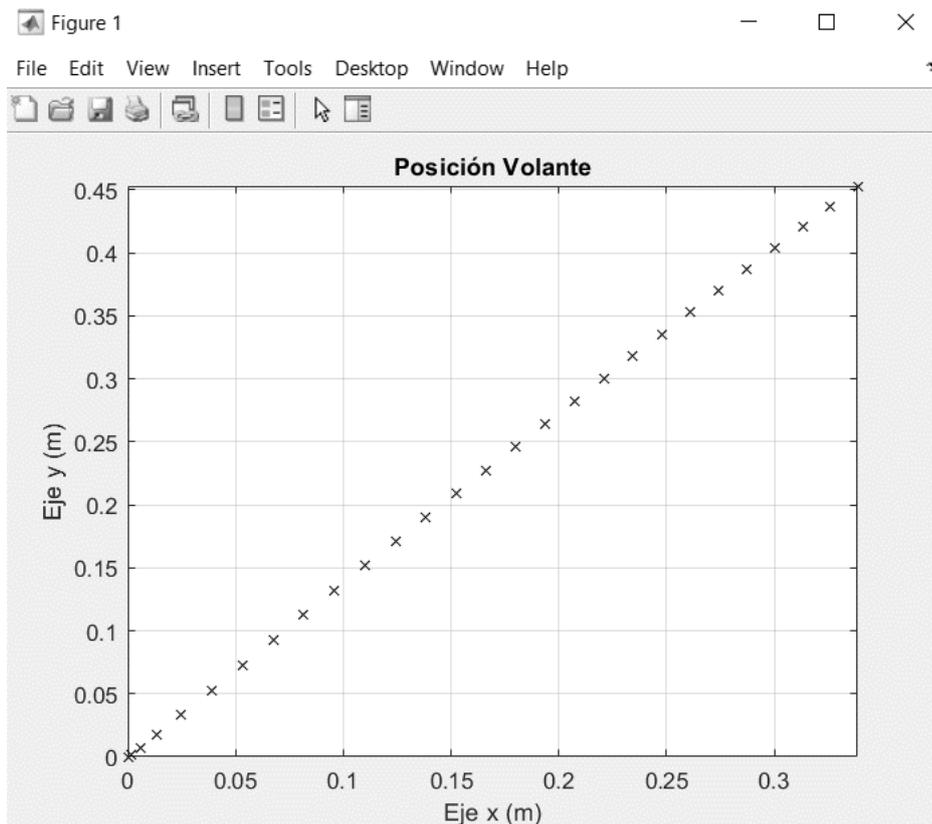


Figura 38. Gráfica de posición. MR. MATLAB.

Se realiza el ajuste por mínimos cuadrados de los datos obtenidos en MATLAB, y se representan en una gráfica.

- La correlación obtenida es: 0,9994062.
- La pendiente obtenida es:  $0,2282 \pm 0,0011$ .
- La ordenada en el origen es:  $-0,0651 \pm 0,0019$ .

Por lo tanto, la velocidad será:  $V = 0,2282 \pm 0,0011$  m/s.

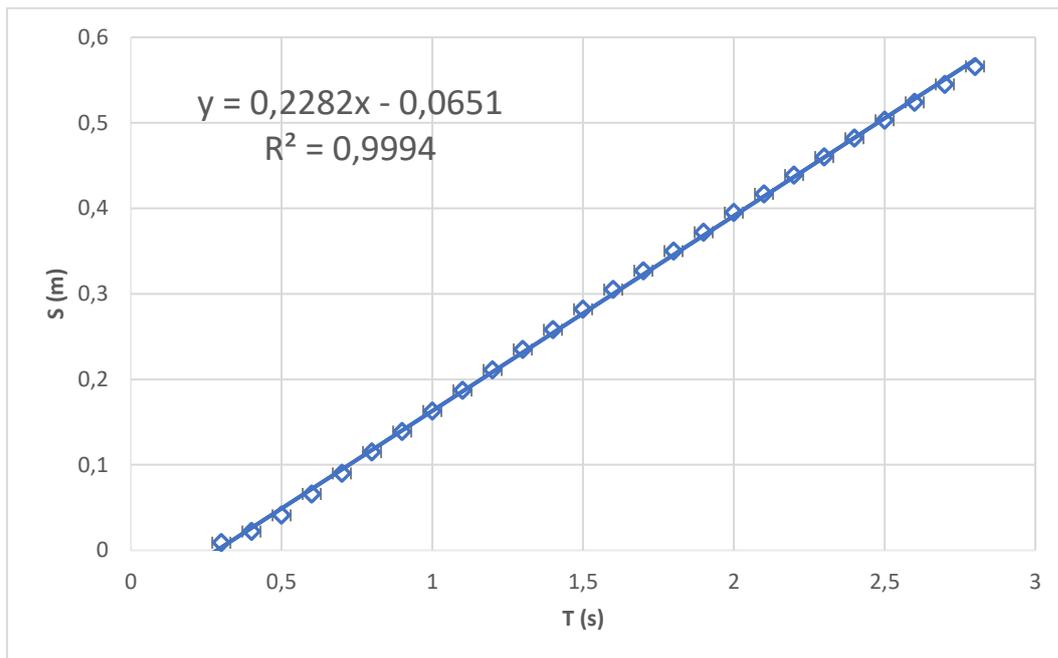


Figura 39. Ajuste por mínimos cuadrados. Posición. MR. MATLAB.

VELOCIDAD DEL VOLANTE

La velocidad disminuye progresivamente con el tiempo. Esto es debido a que existen fuerzas de rozamiento que hacen que el volante desacelere, el colchón de aire creado no es perfecto.

Se muestran en la siguiente tabla, los valores que toma la velocidad en ambos ejes a lo largo del tiempo, y la velocidad obtenida al realizar la raíz cuadrada de la suma de sus componentes.

<b>T ± 0,03 (s)</b>	<b>Vx (m/s)</b>	<b>Vy (m/s)</b>	<b>V (m/s)</b>
0,10	0,01 ± 0,04	0,01 ± 0,04	0,02 ± 0,06
0,20	0,01 ± 0,04	0,01 ± 0,04	0,02 ± 0,06
0,30	0,04 ± 0,05	0,06 ± 0,06	0,07 ± 0,08
0,40	0,08 ± 0,06	0,11 ± 0,07	0,13 ± 0,10
0,50	0,11 ± 0,07	0,16 ± 0,09	0,20 ± 0,11
0,60	0,14 ± 0,08	0,20 ± 0,10	0,24 ± 0,13
0,70	0,14 ± 0,08	0,20 ± 0,10	0,25 ± 0,13
0,80	0,14 ± 0,08	0,20 ± 0,10	0,25 ± 0,13
0,90	0,14 ± 0,08	0,20 ± 0,10	0,24 ± 0,13
1,00	0,14 ± 0,08	0,20 ± 0,10	0,24 ± 0,13
1,10	0,14 ± 0,08	0,19 ± 0,10	0,24 ± 0,13
1,20	0,14 ± 0,08	0,19 ± 0,10	0,24 ± 0,13
1,30	0,14 ± 0,08	0,19 ± 0,10	0,24 ± 0,13
1,40	0,14 ± 0,09	0,19 ± 0,10	0,24 ± 0,13
1,50	0,14 ± 0,08	0,19 ± 0,10	0,23 ± 0,13
1,60	0,14 ± 0,08	0,18 ± 0,10	0,23 ± 0,12
1,70	0,14 ± 0,08	0,18 ± 0,10	0,23 ± 0,12
1,80	0,14 ± 0,08	0,18 ± 0,09	0,23 ± 0,12
1,90	0,14 ± 0,08	0,18 ± 0,09	0,23 ± 0,12
2,00	0,14 ± 0,08	0,18 ± 0,09	0,22 ± 0,12
2,10	0,13 ± 0,08	0,18 ± 0,09	0,22 ± 0,12
2,20	0,13 ± 0,08	0,17 ± 0,09	0,22 ± 0,12
2,30	0,13 ± 0,08	0,17 ± 0,09	0,22 ± 0,12
2,40	0,13 ± 0,08	0,17 ± 0,09	0,21 ± 0,12
2,50	0,13 ± 0,08	0,17 ± 0,09	0,21 ± 0,12
2,60	0,13 ± 0,08	0,17 ± 0,09	0,21 ± 0,12
2,70	0,13 ± 0,08	0,17 ± 0,09	0,21 ± 0,12
2,80	0,13 ± 0,08	0,16 ± 0,09	0,21 ± 0,12

Tabla 2. Velocidad del volante en el tiempo. MR. MATLAB.

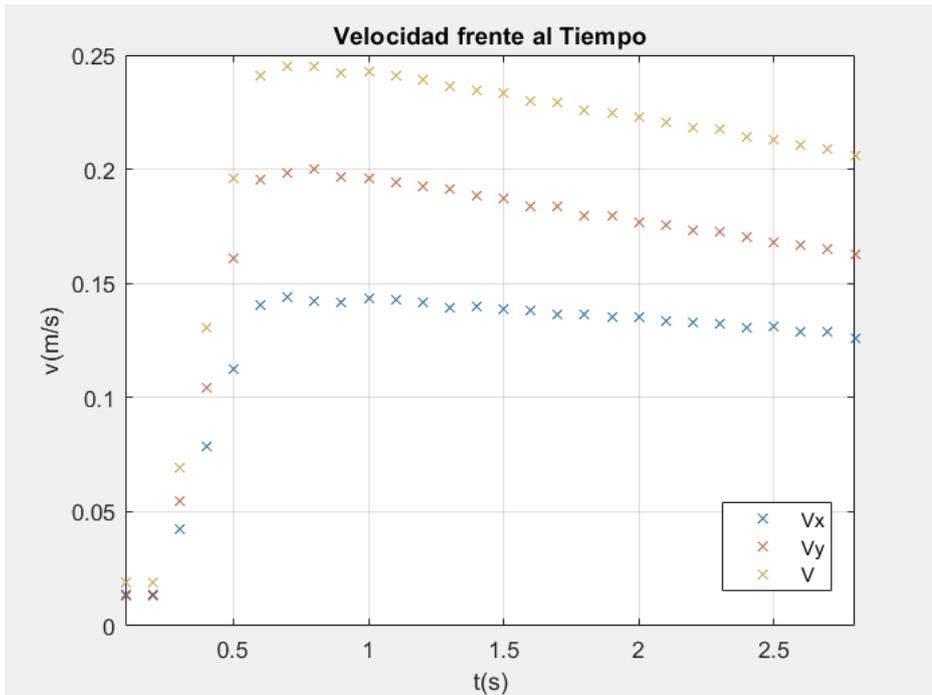


Figura 40. Velocidad frente al tiempo. MR. MATLAB.

Se realiza el ajuste por mínimos cuadrados de los datos obtenidos en MATLAB, y se representan en una gráfica.

- La correlación obtenida es: 0,9992903.
- La pendiente obtenida es: - 0,02059 ± 0,00015.
- La ordenada en el origen es: 0,2638 ± 0,0003.

Por lo tanto, la aceleración será:  $A = - 0,02059 \pm 0,00015 \text{ m/s}^2$ .

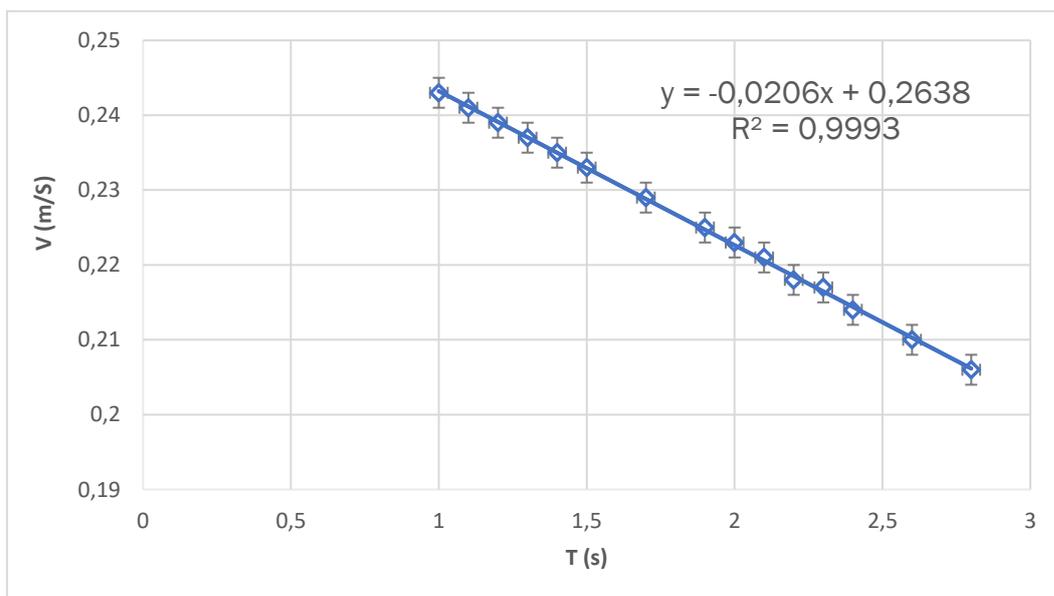


Figura 41. Ajuste por mínimos cuadrados. Velocidad. MR. MATLAB.

ACELERACIÓN DEL VOLANTE

Se puede observar una gran aceleración inicial al ser impulsado el volante con la mano, hasta que alcanza su valor máximo. Después de este instante, la aceleración desciende drásticamente, debido a que cesa el impulso inicial. Luego la aceleración es prácticamente nula, aunque se puede observar como la aceleración,  $A$ , es negativa con valores cercanos a cero. Esto es debido a las fuerzas de rozamiento que frenan el volante por no tratarse de condiciones ideales.

En este caso se muestran las aceleraciones sin sus respectivos errores, debido a que las magnitudes se han calculado a partir de los incrementos de velocidad y estos a su vez de las posiciones. Por lo tanto, los errores se van amplificando y transmitiendo a las siguientes magnitudes. Llegando en el caso de las aceleraciones hasta mostrar un error de  $\pm 2 \text{ m/s}^2$  cuando el valor máximo que alcanza la aceleración es de  $0,6 \text{ m/s}^2$ . En consecuencia, se va a tomar como valor correcto de la aceleración el obtenido anteriormente ajustando por mínimos cuadrados, junto con el error de la pendiente.

$$A = -0,02059 \pm 0,00015 \text{ m/s}^2.$$

Se muestran en la siguiente tabla, los valores que toma la aceleración en ambos ejes a lo largo del tiempo, y la aceleración obtenida al realizar la raíz cuadrada de la suma de sus componentes.

$T \pm 0,03 \text{ (s)}$	$A_x \text{ (m/s}^2\text{)}$	$A_y \text{ (m/s}^2\text{)}$	$A \text{ (m/s}^2\text{)}$
0,10	0,285	0,414	0,502
0,20	0,285	0,414	0,502
0,30	0,285	0,414	0,502
0,40	0,360	0,499	0,615
0,50	0,340	0,562	0,657
0,60	0,285	0,349	0,450
0,70	0,033	0,029	0,044
0,80	-0,020	0,015	0,025
0,90	-0,003	-0,033	-0,033
1,00	0,017	-0,006	0,018
1,10	-0,007	-0,020	-0,021
1,20	-0,012	-0,013	-0,018
1,30	-0,021	-0,016	-0,027
1,40	0,007	-0,027	-0,028
1,50	-0,012	-0,012	-0,017
1,60	-0,008	-0,036	-0,037
1,70	-0,015	0,004	-0,015

1,80	-0,005	-0,042	-0,043
1,90	-0,010	-0,003	-0,011
2,00	0,002	-0,025	-0,025
2,10	-0,019	-0,013	-0,023
2,20	-0,007	-0,026	-0,027
2,30	-0,007	-0,004	-0,008
2,40	-0,015	-0,026	-0,030
2,50	0,007	-0,023	-0,024
2,60	-0,027	-0,012	-0,030
2,70	0,000	-0,018	-0,018
2,80	-0,026	-0,022	-0,034

Tabla 3. Aceleración del volante en el tiempo. MR. MATLAB.

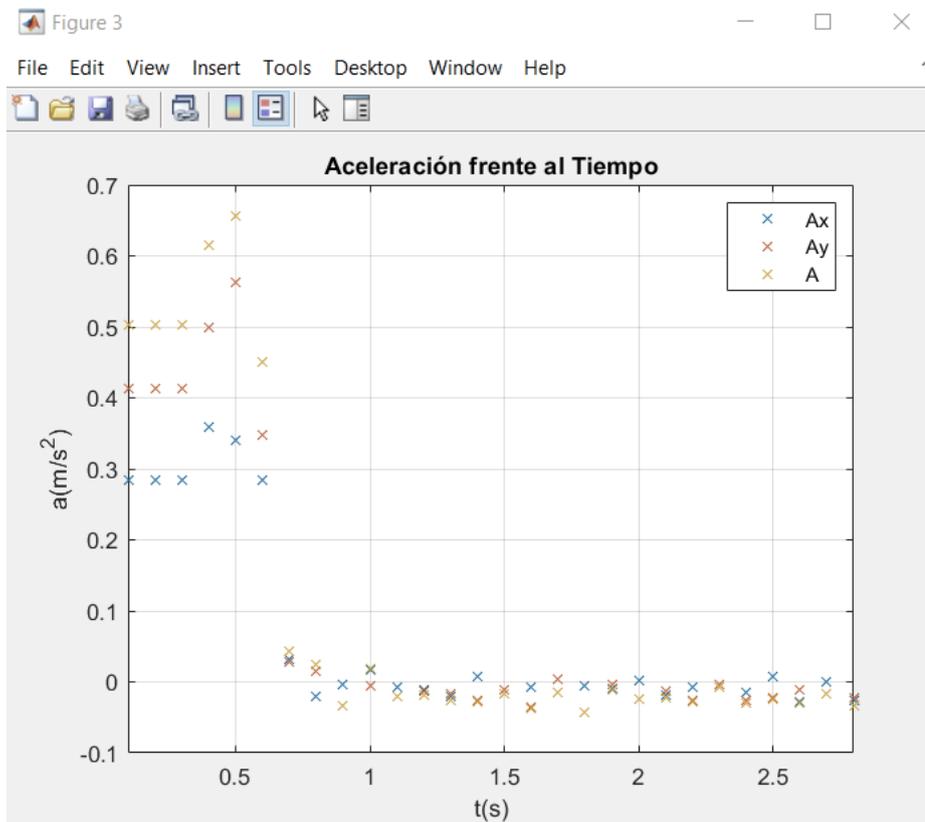


Figura 42. Aceleración frente al tiempo. MR. MATLAB.

Se representan en una gráfica únicamente los valores de la aceleración después del impulso inicial del volante. Para observar con mayor claridad como la fricción producida por fluidos no genera una desaceleración constante, pero estas sí varían en torno a un valor constante. Este valor es la pendiente de la

velocidad,  $A = -0,02059 \pm 0,00015 \text{ m/s}^2$  representado en la gráfica por la recta horizontal naranja.

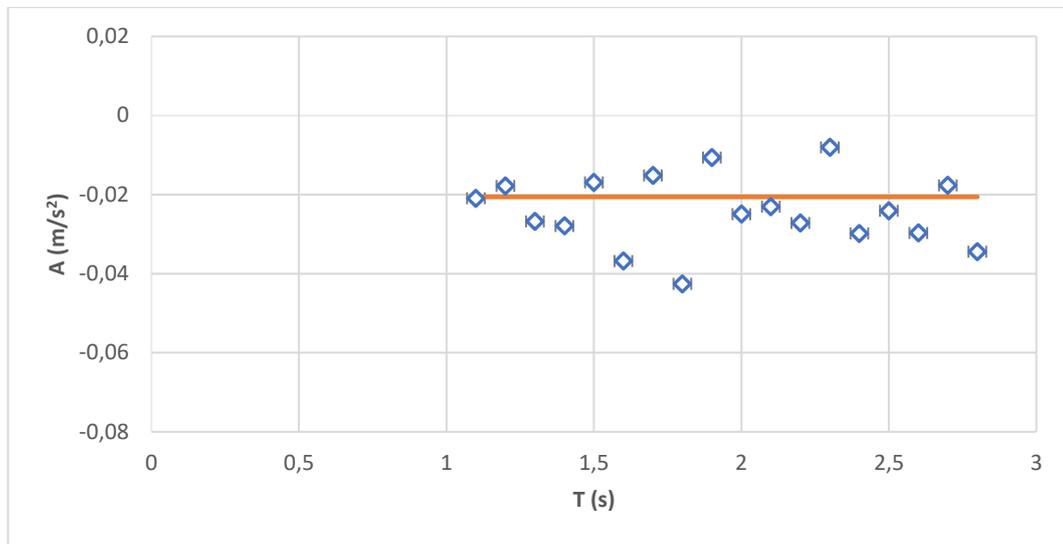
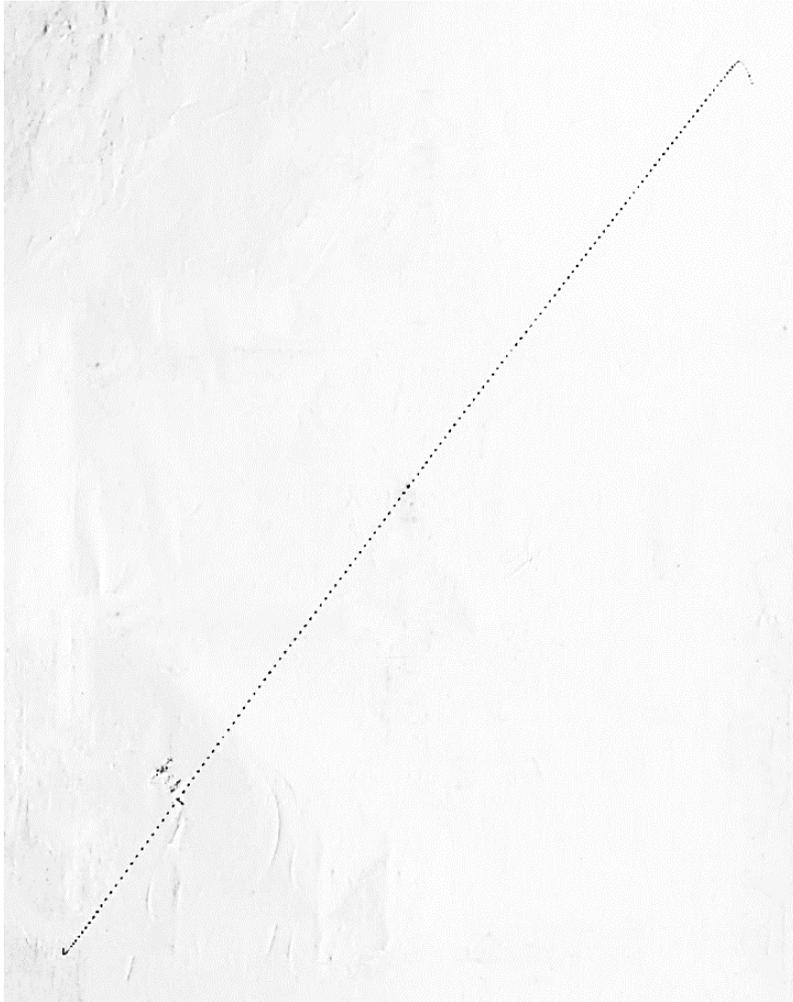


Figura 43. Aceleración frente al tiempo y cte. MR. MATLAB.

## RESULTADO EXPERIMENTAL PAPEL ELECTROCONDUCTOR

A continuación se muestran los resultados obtenidos a partir de los impulsos del electrodo central del volante en el papel electroconductor. La frecuencia con la se realizan los impulsos es 50 Hz. Por lo tanto, entre cada posición marcada hay 0,02 segundos.



*Figura 44. Papel electroconductor. MR*

Se toma el mismo intervalo de tiempo que en Matlab, 0,1 segundos entre pulsaciones para aproximar los datos lo máximo posible. El electrodo realiza marcaciones en el papel cada 0,02 segundos. Se tomarán en cuenta las posiciones en intervalos de 5 en 5. Se medirá entre el inicio de las pulsaciones la distancia entre ellas para así poder hallar la velocidad y la aceleración.

Se puede observar en los datos obtenidos de MATLAB que el movimiento se inicia en el fotograma inicial del video. En consecuencia, las medidas tomadas en el papel electroconductor se iniciarán desde el instante inicial también.

Se toman como valores correctos la velocidad y la aceleración obtenidas ajustando por mínimos cuadrados, gracias al error de la pendiente, por el mismo motivo anterior de la concatenación de errores. Se anotan en una tabla los resultados obtenidos.

T ± 0,02 (s)	S ± 0,001 (m)	V (m/s)	A (m/s <sup>2</sup> )
0,10	0,005	0,10 ± 0,04	0,6 ± 0,8
0,20	0,015	0,10 ± 0,04	0,6 ± 0,8
0,30	0,031	0,16 ± 0,05	0,6 ± 1,0
0,40	0,053	0,22 ± 0,06	0,6 ± 1,3
0,50	0,077	0,24 ± 0,07	0,2 ± 1,4
0,60	0,101	0,24 ± 0,07	0,0 ± 1,4
0,70	0,125	0,24 ± 0,07	-0,1 ± 1,4
0,80	0,148	0,23 ± 0,07	-0,1 ± 1,3
0,90	0,171	0,23 ± 0,07	0,0 ± 1,3
1,00	0,194	0,23 ± 0,07	0,0 ± 1,3
1,10	0,216	0,23 ± 0,07	-0,1 ± 1,3
1,20	0,239	0,23 ± 0,07	0,0 ± 1,3
1,30	0,261	0,23 ± 0,07	0,0 ± 1,3
1,40	0,283	0,22 ± 0,06	-0,1 ± 1,3
1,50	0,305	0,22 ± 0,06	0,0 ± 1,3
1,60	0,327	0,22 ± 0,06	0,0 ± 1,3
1,70	0,349	0,22 ± 0,06	-0,1 ± 1,3
1,80	0,370	0,22 ± 0,06	0,0 ± 1,3
1,90	0,392	0,22 ± 0,06	0,0 ± 1,3
2,00	0,413	0,21 ± 0,06	-0,1 ± 1,3
2,10	0,434	0,21 ± 0,06	0,0 ± 1,2
2,20	0,455	0,21 ± 0,06	0,0 ± 1,2
2,30	0,475	0,21 ± 0,06	-0,1 ± 1,2
2,40	0,496	0,21 ± 0,06	0,0 ± 1,2
2,50	0,516	0,20 ± 0,06	-0,1 ± 1,2
2,60	0,535	0,19 ± 0,06	-0,1 ± 1,2
2,70	0,554	0,19 ± 0,06	0,0 ± 1,2
2,80	0,572	0,18 ± 0,06	-0,1 ± 1,2

Tabla 4. Datos del volante en el tiempo. MR. Papel electroconductor

## POSICIÓN

Se realiza el ajuste por mínimos cuadrados de las posiciones obtenidas con el papel electroconductor, y se representan en una gráfica.

- La correlación obtenida es: 0,9993697
- La pendiente obtenida es: 0,2190 ± 0,0011.
- La ordenada en el origen es: - 0,0278 ± 0,0018.

Por lo tanto, la velocidad será:  $V = 0,2190 \pm 0,0011$  m/s.

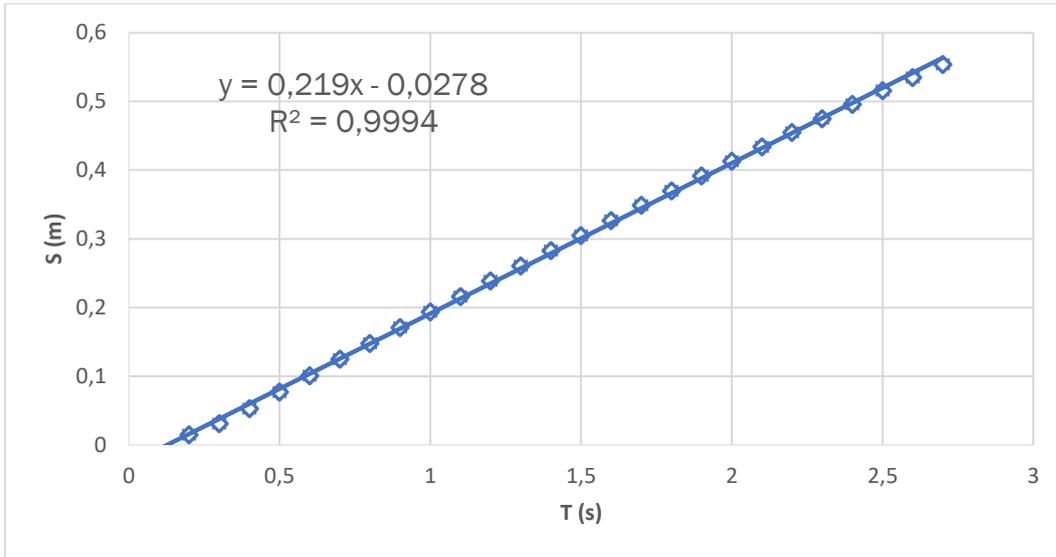


Figura 45. Ajustes mínimos cuadrados. Posición frente al tiempo. MR. Papel electroconductor.

## VELOCIDAD

Se realiza el ajuste por mínimos cuadrados de las velocidades obtenidas con el papel electroconductor, y se representan en una gráfica.

- La correlación obtenida es: 0,9981481
- La pendiente obtenida es:  $- 0,0194 \pm 0,0003$ .
- La ordenada en el origen es:  $- 0,2492 \pm 0,0005$ .

Por lo tanto, la aceleración será:  $A = - 0,0194 \pm 0,0003$  m/s<sup>2</sup>.

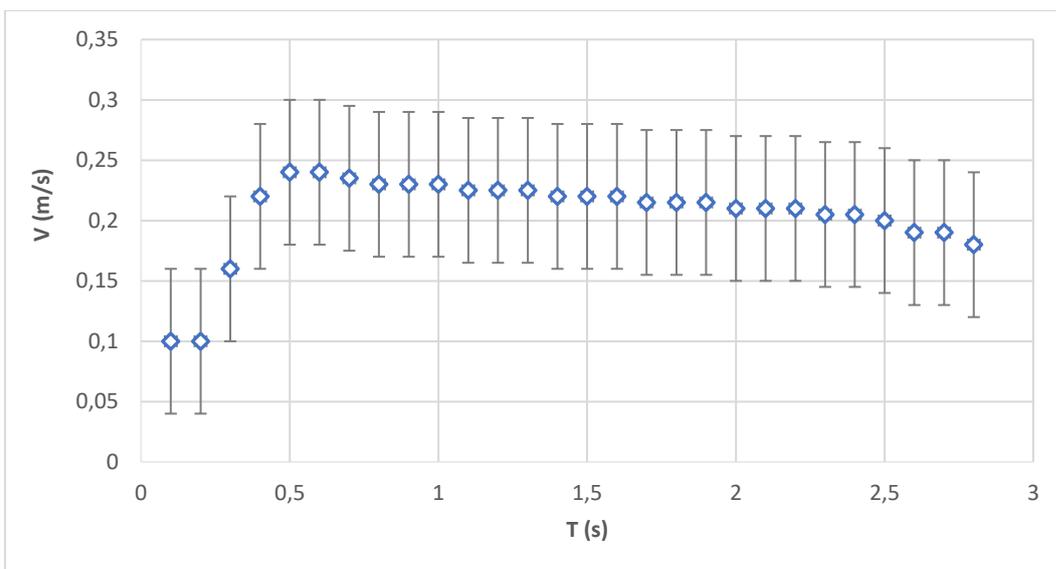


Figura 46. Velocidad frente al tiempo. MR. Papel electroconductor.

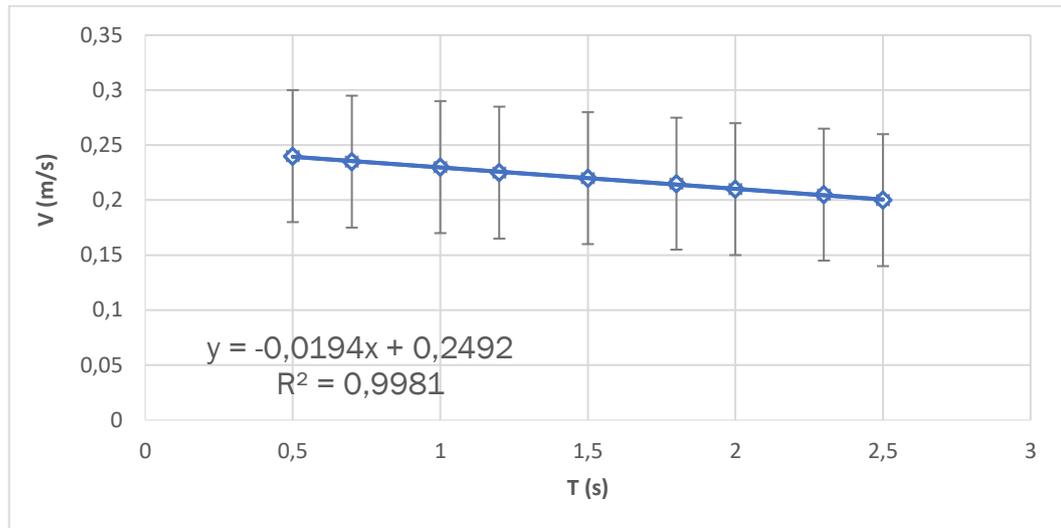


Figura 47. Ajustes mínimos cuadrados. Velocidad frente al tiempo. Papel electroconductor.

### ACELERACIÓN

Se representan en una gráfica las aceleraciones obtenidas con el papel electroconductor. Se puede observar un primer impulso inicial por el empuje del volante con la mano. Después cesa esa aceleración inicial y varía en torno a un valor constante negativo próximo a cero, haciendo que el volante desacelere al no tratarse de condiciones ideales.

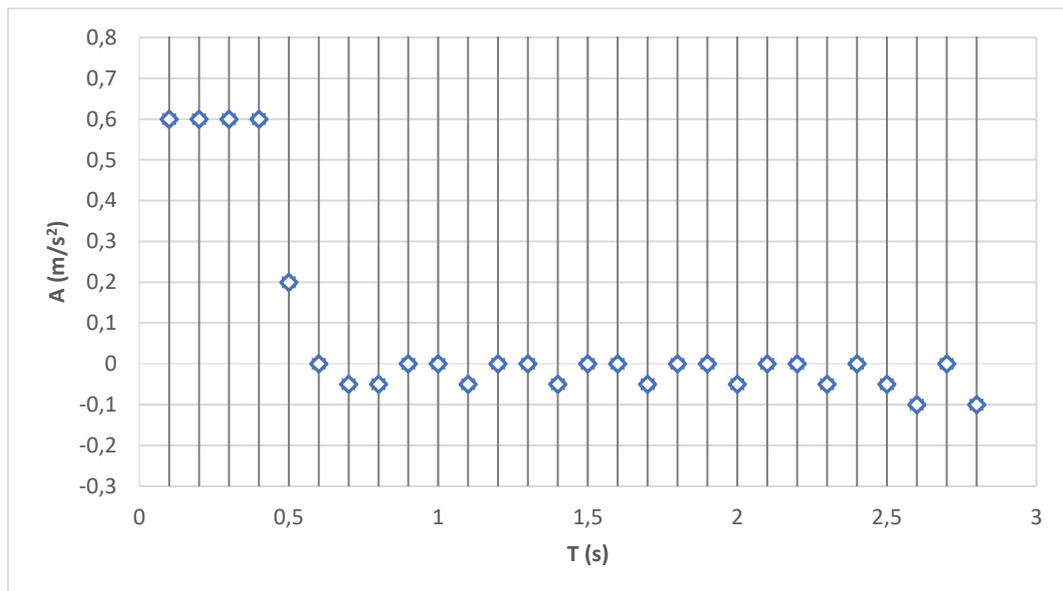


Figura 48. Aceleración frente al tiempo. MR. Papel electroconductor.

Se representan en una gráfica únicamente los valores de la aceleración después del impulso inicial del volante, observando solo el movimiento rectilíneo uniforme. Para mostrar con mayor claridad como la fricción producida por fluidos no genera una desaceleración constante, pero estas sí varían en torno a un valor constante. Este valor es la pendiente de la velocidad,  $A = -0,0194 \pm 0,0003 \text{ m/s}^2$  representado en la gráfica por la recta horizontal naranja.

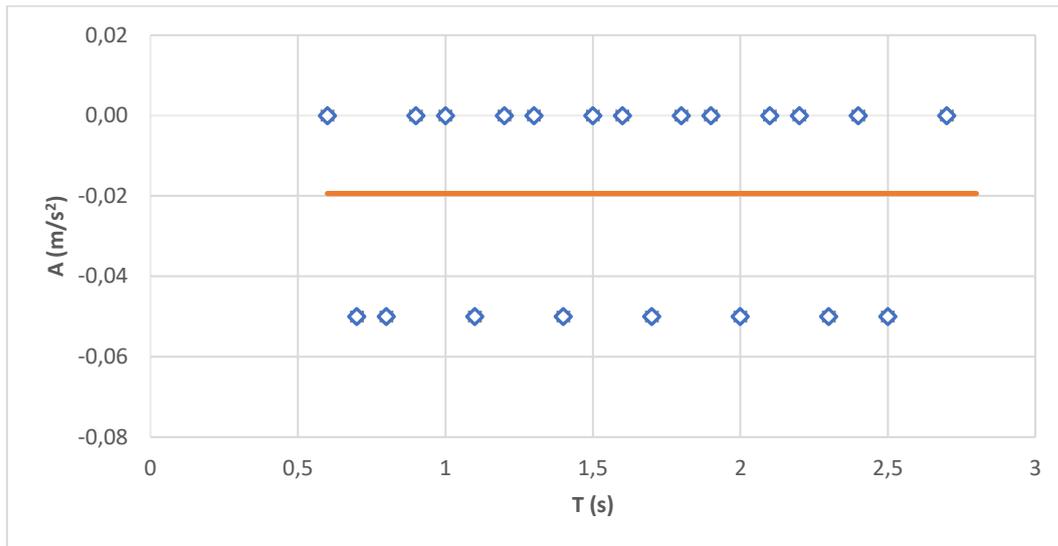


Figura 49. Aceleración frente al tiempo y cte. MR. Papel electroconductor.

### COMPARACIÓN

En primer lugar, se realiza la comprobación teórica observando que el colchón de aire creado no es ideal. Existen fuerzas de rozamiento que hacen que la velocidad del volante disminuya progresivamente. La distancia entre puntos consecutivos no es constante, y después de alcanzar la velocidad máxima, donde esta distancia es máxima también, se reduce con el tiempo.

En segundo lugar, se comparan los datos experimentales obtenidos a partir del papel electroconductor y de MATLAB. Estos datos no van a ser exactos, pero la velocidad y la aceleración si debieran ser aproximadas.

No existe esta exactitud por diferentes circunstancias externas:

- El instante inicial de ambos movimientos no es exactamente el mismo, se ha aproximado lo máximo posible, pero puede haber errores de décimas de segundo entre ambos movimientos.
- Los ejes de coordenadas se han aproximado, pero no son exactos, por ello se ha preferido medir la distancia entre puntos que se considera más precisa.
- Pueden existir errores de  $\pm 1$  mm en la medición del papel electroconductor por el instrumento de medición utilizado.

#### Interpretación de los resultados:

Se comparan ambos resultados experimentales:

El desplazamiento final observado en MATLAB es  $0,566 \pm 0,003$  m, muy similar al obtenido por el papel electroconductor  $0,572 \pm 0,001$  m. El desplazamiento a lo largo del tiempo también es similar.

Las velocidades tomadas por el volante a lo largo del tiempo son parecidas en ambos experimentos, en el papel electroconductor se observa una velocidad de  $0,2190 \pm 0,0011$  m/s, similar a velocidad  $0,2282 \pm 0,0011$  m/s observada en MATLAB.

Las aceleraciones tomadas por el volante a lo largo del tiempo son similares en ambas, pero con algunas diferencias:

- Por una parte, las aceleraciones observadas en MATLAB, a partir del máximo, son negativas siempre y varían próximas a cero con valor prácticamente constante, haciendo que disminuya la velocidad de manera progresiva.

- Por la otra parte, las aceleraciones observadas en el papel electroconductor tienen cambios más bruscos, debido a la baja precisión del instrumento de medida, y a trabajar a bajas velocidades.
- En el papel electroconductor se observa una aceleración de  $-0,0194 \pm 0,0003 \text{ m/s}^2$ , similar a aceleración  $-0,02059 \pm 0,00015 \text{ m/s}^2$  observada en MATLAB.

Al comparar ambos resultados se observa un resultado satisfactorio. Hay pequeñas diferencias en ambas prácticas, pero son asumibles debido a la inexactitud por las circunstancias externas mencionadas anteriormente.

### 5.3. MOVIMIENTO RECTILÍNEO UNIFORMEMENTE VARIADO

#### MARCO TEÓRICO

El movimiento rectilíneo uniformemente variado (M.R.U.V.) acelerado o desacelerado es un tipo de movimiento en el que un objeto se desplaza a lo largo de una línea recta con cierta aceleración constante, por lo tanto, la velocidad varía con el tiempo.

La posición de un objeto que sigue un movimiento rectilíneo uniformemente acelerado o desacelerado viene dada por la siguiente ecuación:

$$s(t) = s_0 + v_0 \cdot t + \frac{1}{2} a \cdot t^2 \quad (2)$$

$s_0 \rightarrow$  Posición inicial,  $v_0 \rightarrow$  Velocidad inicial, ambos para  $t = 0$  s, y  $a \rightarrow$  Aceleración de valor constante.

Al representar los valores que toma  $s$  frente al tiempo se obtiene una parábola.

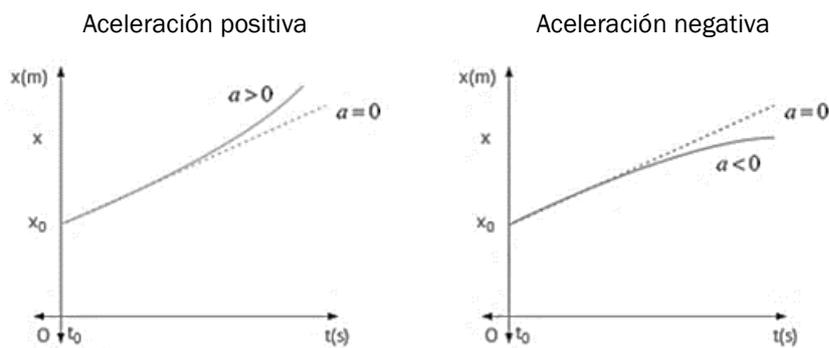


Figura 50. Gráfica  $x - t$  en M.R.U.V. [3]

La velocidad de un objeto que sigue un movimiento rectilíneo uniforme viene dada por la siguiente ecuación:

$$v(t) = v_0 + a \cdot t \quad (3)$$

Al representar los valores que toma  $v$  frente al tiempo se obtiene una línea recta de pendiente positiva o negativa en función del signo de la aceleración.

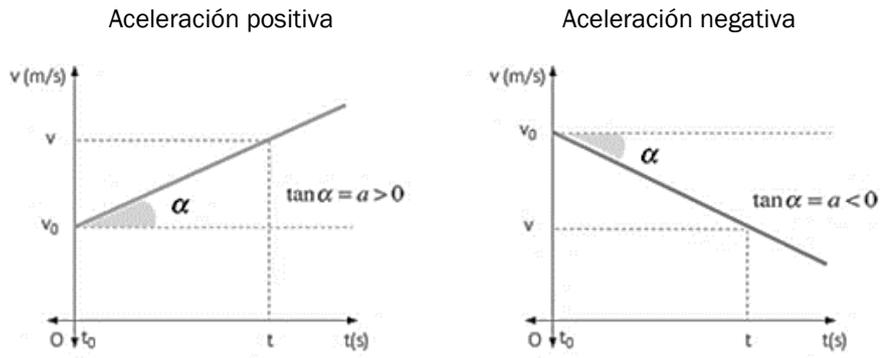


Figura 51. Gráfica  $v - t$  en M.R.U.V. [3]

### EXPLICACIÓN DEL M.R.U.V. REALIZADO

Se coloca un hilo de masa despreciable unido al volante y a un peso colgante de 18 gramos. Si se acciona el interruptor se crea el colchón de aire que reduce prácticamente a cero la fuerza de fricción. Sin esta fuerza de rozamiento que se oponga a la tensión del hilo producida por el peso, el volante comienza a moverse en su dirección incrementando la velocidad proporcionalmente al tiempo.

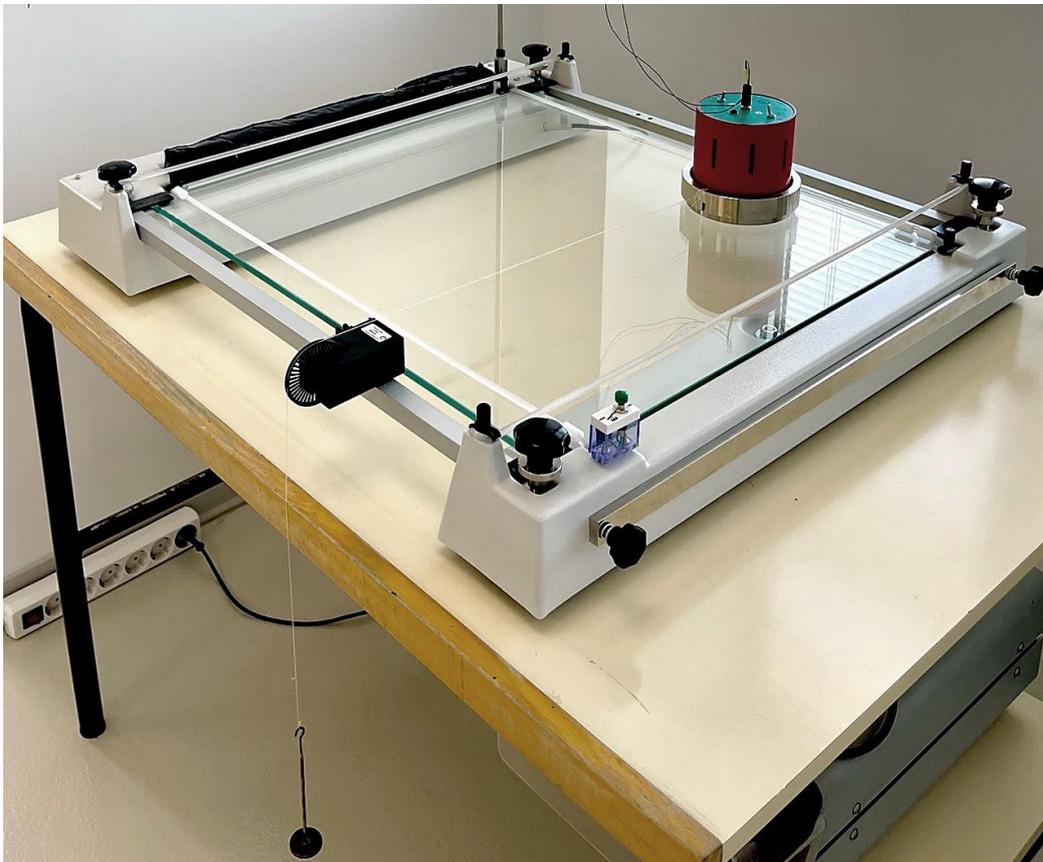


Figura 52. Práctica MRUV

Se coloca el papel electroconductor entre el volante y el cristal de la mesa neumática, a su vez se posiciona el trípode con la cámara en la parte superior para poder captar el mismo movimiento de las dos formas y así poder compararlo.

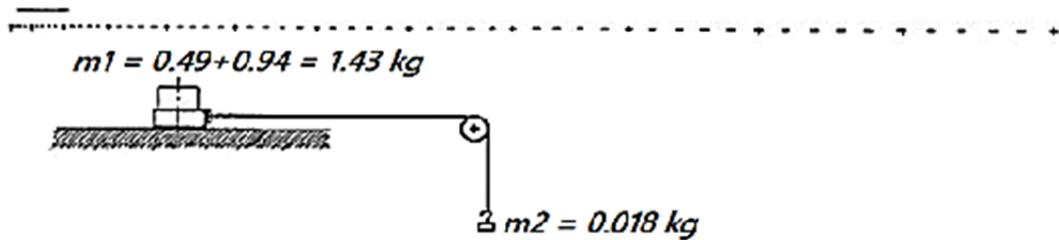


Figura 53. Práctica MRUV

A continuación, se inicia la grabación de la cámara y se acciona el interruptor, que genera los impulsos del electrodo en el papel electroconductor, pudiendo así comparar ambos resultados.

Se realiza el movimiento rectilíneo uniformemente variado (acelerado) del volante. En la figura 54 se puede observar el lanzamiento realizado, se observa en la zona superior el fotograma inicial y en la inferior el final del movimiento.

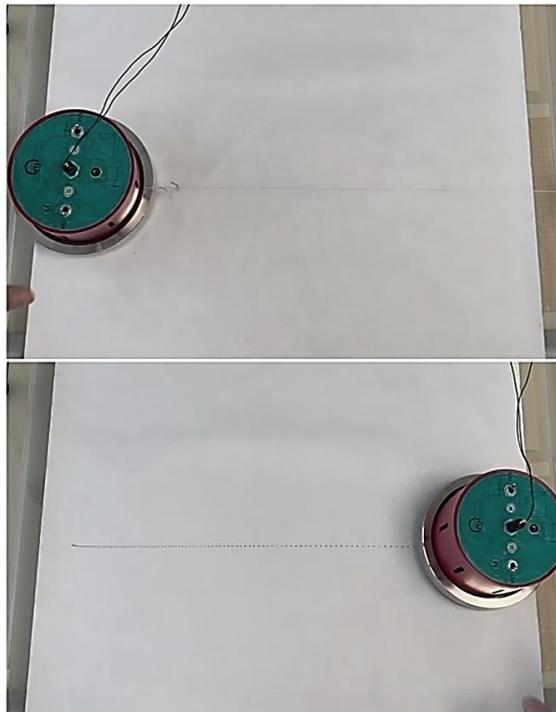


Figura 54. Video MRUV

## RESULTADO TEÓRICO

Se realiza el diagrama de fuerzas del sistema de esta práctica, adoptando varias consideraciones ideales para simplificar el análisis:

- El comportamiento de la polea se considera ideal, con momento de inercia y fricción despreciables. Por lo tanto, la tensión en el hilo será igual a ambos lados del contacto con la polea.
- El hilo se considera inextensible y de masa despreciable, de manera que la tensión es la misma en todos sus puntos.
- Se considera el hilo rígido, inelástico. La aceleración que tiene el volante es igual a la del peso.
- No se tiene en cuenta la fricción entre el volante y el colchón de aire, que afecta a la aceleración del sistema. Se considera la fuerza de rozamiento nula.

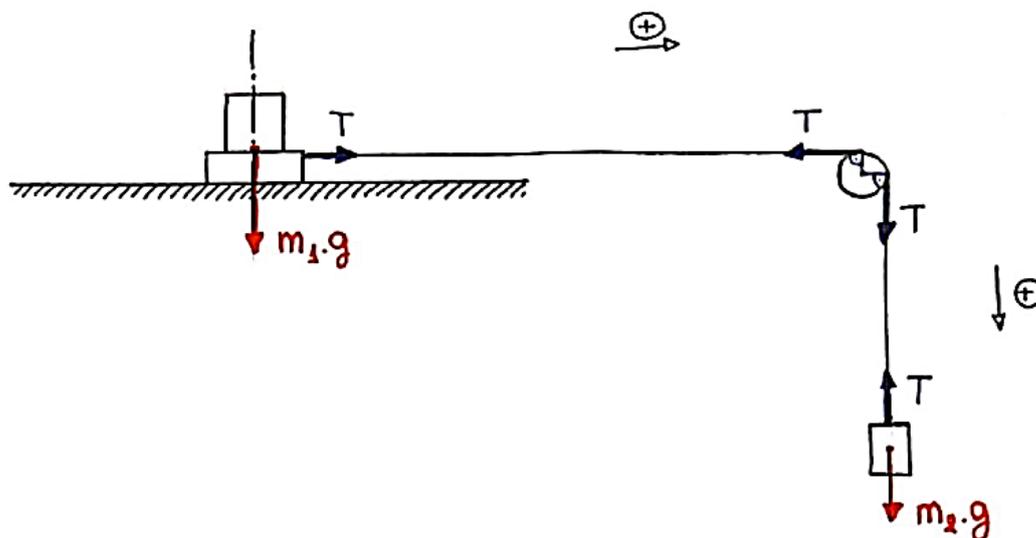


Figura 55. Diagrama de fuerzas. MRUV

Se realiza el sumatorio de fuerzas horizontal del sólido 1, siendo positivo el sentido derecho:

$$\sum \vec{F}_H = m \cdot \vec{a}$$

$$\vec{T} = m_1 \cdot \vec{a}_1$$

Se realiza el sumatorio de fuerzas verticales del sólido 2, siendo positivo el sentido hacia abajo:

$$\sum \vec{F}_V = m \cdot \vec{a}$$

$$m_2 \cdot \vec{g} - \vec{T} = m_2 \cdot \vec{a}_2$$

Partiendo de las premisas:  $\vec{a} = \vec{a}_1 = \vec{a}_2$

Igualando las ecuaciones anteriores, se obtiene la ecuación de la aceleración teórica buscada:

$$m_2 \cdot \vec{g} - m_1 \cdot \vec{a} = m_2 \cdot \vec{a}$$

$$\vec{a} = \frac{m_2 \cdot \vec{g}}{m_1 + m_2}$$

Se sustituye cada parámetro con su valor real.

$$\vec{a} = \frac{0,018 \cdot 9,81}{1,430 + 0,018} = 0,1219 \text{ m/s}^2$$

Por último, partiendo de los errores iniciales por la precisión del aparato de medida en las masas de ambos sólidos.

$$m_1 = 1,430 \pm 0,001 \text{ Kg}$$

$$m_2 = 0,018 \pm 0,001 \text{ Kg}$$

$$g = 9,81 \pm 0,01 \text{ m/s}^2$$

Se calcula el error de la aceleración teórica:

$$\varepsilon(a) = \left| \frac{\partial a}{\partial m_1} \right| \cdot \varepsilon(m_1) + \left| \frac{\partial a}{\partial m_2} \right| \cdot \varepsilon(m_2) + \left| \frac{\partial a}{\partial g} \right| \cdot \varepsilon(g)$$

$$\varepsilon(a) = \left| -\frac{m_2 \cdot g}{(m_1 + m_2)^2} \right| \cdot \varepsilon(m_1) + \left| \frac{m_1 \cdot g}{(m_1 + m_2)^2} \right| \cdot \varepsilon(m_2) + \left| \frac{m_2}{m_1 + m_2} \right| \cdot \varepsilon(g)$$

$$\varepsilon(a) = \frac{9,81 \cdot (0,018 + 1,430)}{(1,430 + 0,018)^2} \cdot 0,001 + \frac{0,018}{1,430 + 0,018} \cdot 0,01 = 0,0069$$

Se obtiene un valor teórico para la aceleración seguida por el volante en el movimiento rectilíneo uniformemente acelerado de  $0,122 \pm 0,007 \text{ m/s}^2$ .

A continuación, se van a representar los datos del movimiento rectilíneo uniformemente variado obtenidos en MATLAB, a partir del video, y los del papel electroconductor, gracias al electrodo central.

#### RESULTADO EXPERIMENTAL MATLAB

Los datos obtenidos al ejecutar el programa con el video del movimiento rectilíneo uniformemente variado acelerado son los siguientes:

#### POSICIÓN DEL VOLANTE

Se ha colocado el video para que el movimiento se realizase sobre el eje de las X, y así los datos de las Y sean nulos. Pero la grabación no es exacta, y existen un ligero movimiento en eje de las Y. Se toma "S (m)" como el módulo del desplazamiento total desde el origen de coordenadas.

T ± 0,03 (s)	X ± 0,002 (m)	Y ± 0,002 (m)	S (m)
0,10	0,000	0,005	0,005 ± 0,002
0,20	0,000	0,004	0,004 ± 0,002
0,30	0,003	0,004	0,005 ± 0,003
0,40	0,007	0,004	0,008 ± 0,003
0,50	0,013	0,003	0,013 ± 0,002
0,60	0,020	0,003	0,020 ± 0,002
0,70	0,028	0,003	0,028 ± 0,002
0,80	0,038	0,003	0,038 ± 0,002
0,90	0,049	0,003	0,049 ± 0,002
1,00	0,061	0,003	0,061 ± 0,002
1,10	0,075	0,003	0,075 ± 0,002
1,20	0,090	0,003	0,090 ± 0,002
1,30	0,106	0,003	0,106 ± 0,002
1,40	0,124	0,003	0,124 ± 0,002
1,50	0,143	0,003	0,143 ± 0,002
1,60	0,163	0,003	0,163 ± 0,002
1,70	0,184	0,003	0,184 ± 0,002
1,80	0,207	0,003	0,207 ± 0,002
1,90	0,231	0,003	0,231 ± 0,002
2,00	0,256	0,002	0,256 ± 0,002
2,10	0,282	0,002	0,282 ± 0,002
2,20	0,310	0,002	0,310 ± 0,002
2,30	0,339	0,001	0,339 ± 0,002
2,40	0,368	0,001	0,368 ± 0,002
2,50	0,397	0,000	0,397 ± 0,002

Tabla 5. Posición del volante en el tiempo. MRUV. MATLAB.

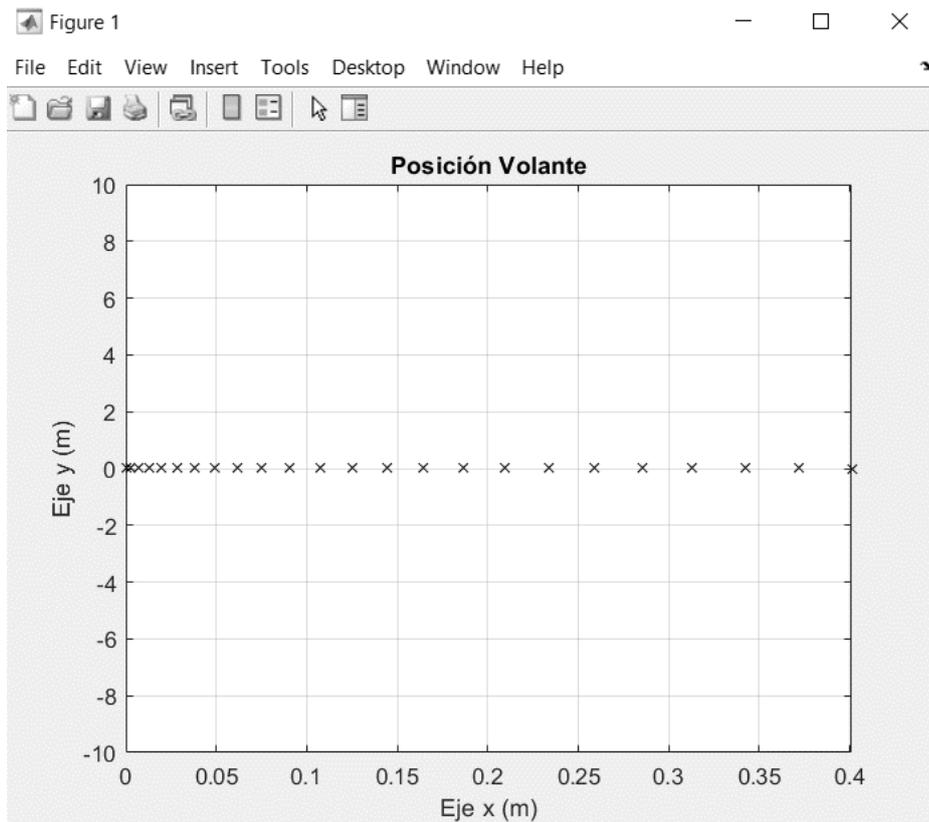


Figura 56. Gráfica de posición. MRUV. MATLAB.

Se realiza el ajuste por mínimos cuadrados de los datos obtenidos en MATLAB, y se representan en una gráfica el desplazamiento frente al tiempo.

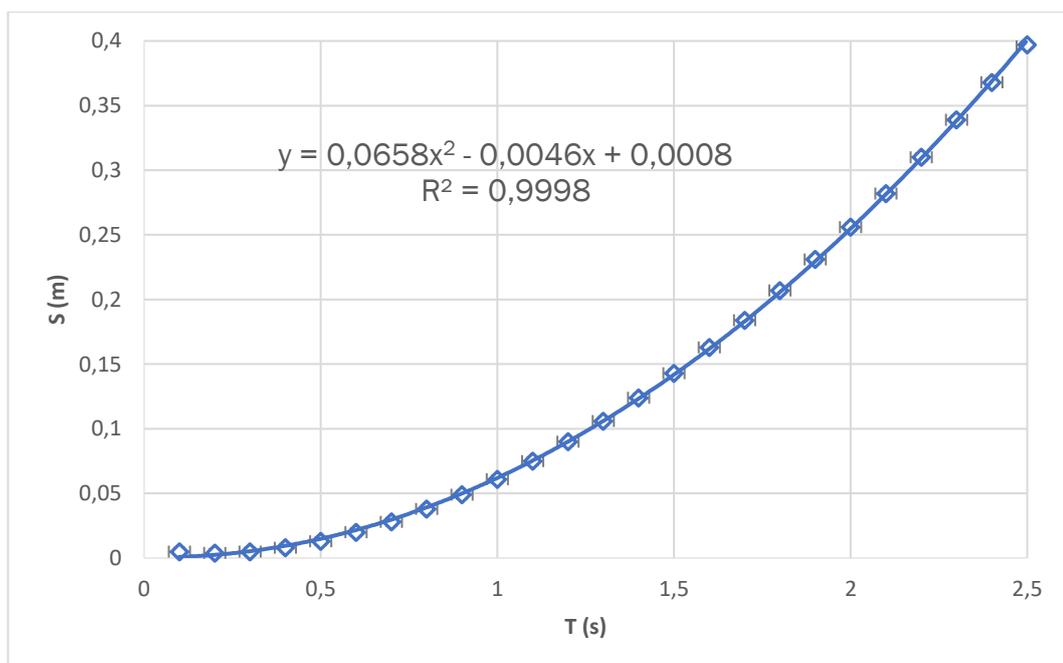


Figura 57. Ajuste por mínimos cuadrados. Posición. MRUV. MATLAB.

### VELOCIDAD DEL VOLANTE

La velocidad inicial es nula, y aumenta progresivamente con el tiempo, debido a la acción de la tensión del hilo, hasta alcanzar el valor máximo. Este se obtiene justo antes de que el volante choque con las bandas elásticas que delimitan la mesa neumática.

Como se ha mencionado anteriormente, este movimiento se realiza prácticamente en el eje de las X, por lo que la velocidad en Y es próxima a cero.

Se muestran en la siguiente tabla, los valores que toma la velocidad en ambos ejes a lo largo del tiempo, y la velocidad obtenida al realizar la raíz cuadrada de la suma de sus componentes.

<b>T ± 0,03 (s)</b>	<b>Vx (m/s)</b>	<b>Vy ± 0,04 (m/s)</b>	<b>V (m/s)</b>
0,10	0,01 ± 0,04	-0,01	0,01 ± 0,06
0,20	0,01 ± 0,04	-0,01	0,01 ± 0,06
0,30	0,02 ± 0,05	0,00	0,02 ± 0,06
0,40	0,04 ± 0,05	0,00	0,04 ± 0,06
0,50	0,06 ± 0,06	0,00	0,06 ± 0,06
0,60	0,07 ± 0,06	0,00	0,07 ± 0,06
0,70	0,09 ± 0,07	0,00	0,09 ± 0,07
0,80	0,10 ± 0,07	0,00	0,10 ± 0,07
0,90	0,11 ± 0,07	0,00	0,11 ± 0,07
1,00	0,12 ± 0,08	0,00	0,12 ± 0,08
1,10	0,14 ± 0,08	0,00	0,14 ± 0,08
1,20	0,15 ± 0,09	0,00	0,15 ± 0,09
1,30	0,17 ± 0,09	0,00	0,17 ± 0,09
1,40	0,18 ± 0,09	0,00	0,18 ± 0,09
1,50	0,19 ± 0,10	0,00	0,19 ± 0,10
1,60	0,20 ± 0,10	0,00	0,20 ± 0,10
1,70	0,22 ± 0,11	0,00	0,22 ± 0,11
1,80	0,23 ± 0,11	0,00	0,23 ± 0,11
1,90	0,24 ± 0,11	0,00	0,24 ± 0,11
2,00	0,25 ± 0,11	0,00	0,25 ± 0,12
2,10	0,26 ± 0,12	0,00	0,26 ± 0,12
2,20	0,28 ± 0,12	0,00	0,28 ± 0,12
2,30	0,29 ± 0,13	0,00	0,29 ± 0,13
2,40	0,30 ± 0,13	-0,01	0,30 ± 0,13
2,50	0,29 ± 0,13	-0,01	0,29 ± 0,13

Tabla 6. Velocidad del volante en el tiempo. MRUV. MATLAB.

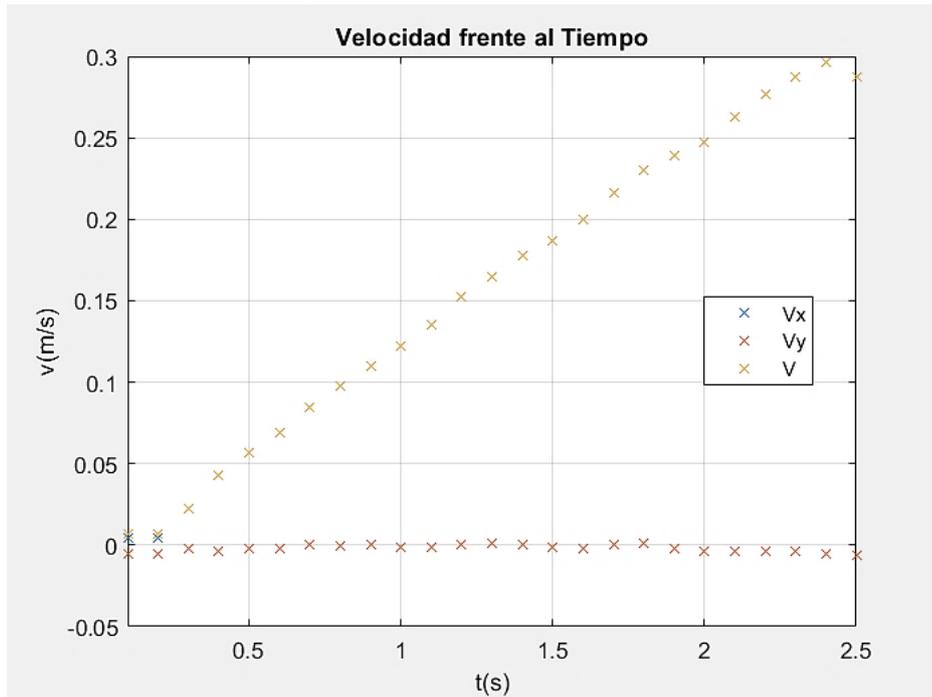


Figura 58. Velocidad frente al tiempo. MRUV. MATLAB.

Se realiza el ajuste por mínimos cuadrados de los datos obtenidos en MATLAB, y se representan en una gráfica.

- La correlación obtenida es: 0,99916338.
- La pendiente obtenida es:  $0,1288 \pm 0,0009$ .
- La ordenada en el origen es:  $-0,0059 \pm 0,0012$ .

Por lo tanto, la aceleración será:  $A = 0,1288 \pm 0,0009 \text{ m/s}^2$ .

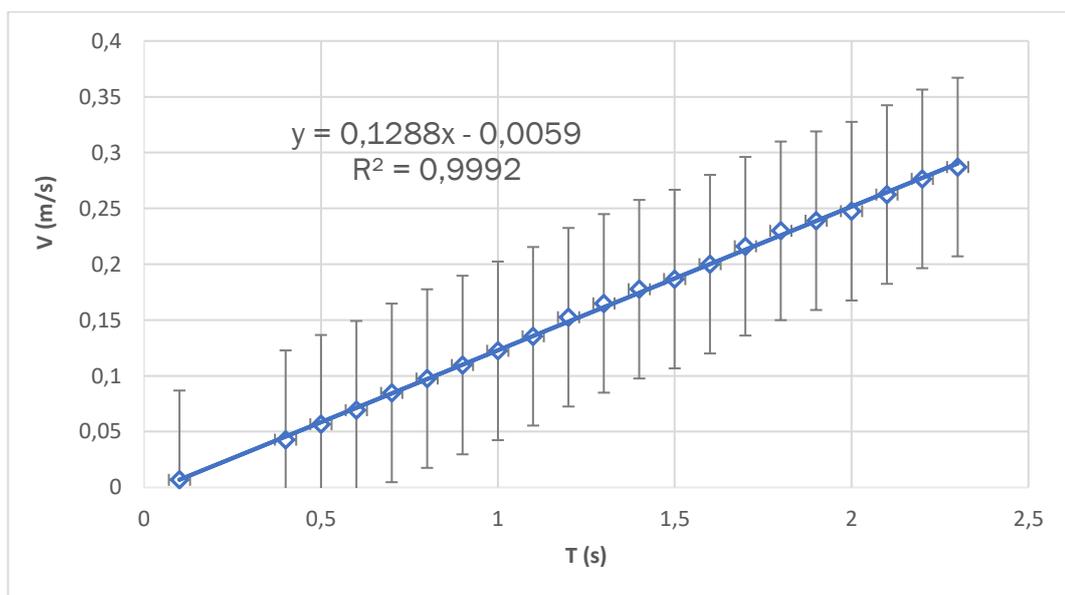


Figura 59. Ajuste por mínimos cuadrados. Velocidad. MRUV. MATLAB.

### ACELERACIÓN DEL VOLANTE

La aceleración en X,  $A_x$ , varía en torno a un valor constante, hasta el instante final donde se produce el choque con las bandas elásticas que hacen frenar el volante, por lo que la aceleración disminuye drásticamente y pasa a ser negativa.

Igual que en las velocidades, la aceleración en el eje de las Y,  $A_y$ , varía en torno a cero.

En este caso se muestran las aceleraciones sin sus respectivos errores, debido a que las magnitudes se han calculado a partir de los incrementos de velocidad y estos a su vez de las posiciones. Por lo tanto, los errores se van amplificando y transmitiendo a las siguientes magnitudes. Llegando en el caso de las aceleraciones hasta mostrar un error de  $\pm 1,3 \text{ m/s}^2$  cuando el valor máximo que alcanza la aceleración es de aproximadamente  $0,2 \text{ m/s}^2$ . En consecuencia, se va a tomar como valor correcto de la aceleración el obtenido anteriormente ajustando por mínimos cuadrados, junto con el error de la pendiente.

$$A = 0,1288 \pm 0,0009 \text{ m/s}^2.$$

Se muestran en la siguiente tabla, los valores que toma la aceleración en ambos ejes a lo largo del tiempo, y la aceleración obtenida al realizar la raíz cuadrada de la suma de sus componentes.

$T \pm 0,03$ (s)	$A_x \text{ (m/s}^2\text{)}$	$A_y \text{ (m/s}^2\text{)}$	$A \text{ (m/s}^2\text{)}$
0,10	0,183	0,030	0,185
0,20	0,183	0,030	0,185
0,30	0,183	0,030	0,185
0,40	0,200	-0,012	0,200
0,50	0,139	0,011	0,140
0,60	0,126	0,002	0,126
0,70	0,156	0,023	0,158
0,80	0,128	-0,007	0,128
0,90	0,121	0,006	0,122
1,00	0,127	-0,013	0,128
1,10	0,130	0,001	0,130
1,20	0,171	0,013	0,171
1,30	0,124	0,009	0,124
1,40	0,127	-0,007	0,127
1,50	0,091	-0,015	0,092
1,60	0,133	-0,008	0,134
1,70	0,161	0,020	0,162

1,80	0,138	0,008	0,138
1,90	0,091	-0,032	0,096
2,00	0,085	-0,011	0,086
2,10	0,149	-0,003	0,149
2,20	0,140	0,002	0,141
2,30	0,106	-0,004	0,106
2,40	0,092	-0,013	0,093
2,50	-0,091	-0,013	-0,092

Tabla 7. Aceleración del volante en el tiempo. MRUV. MATLAB.

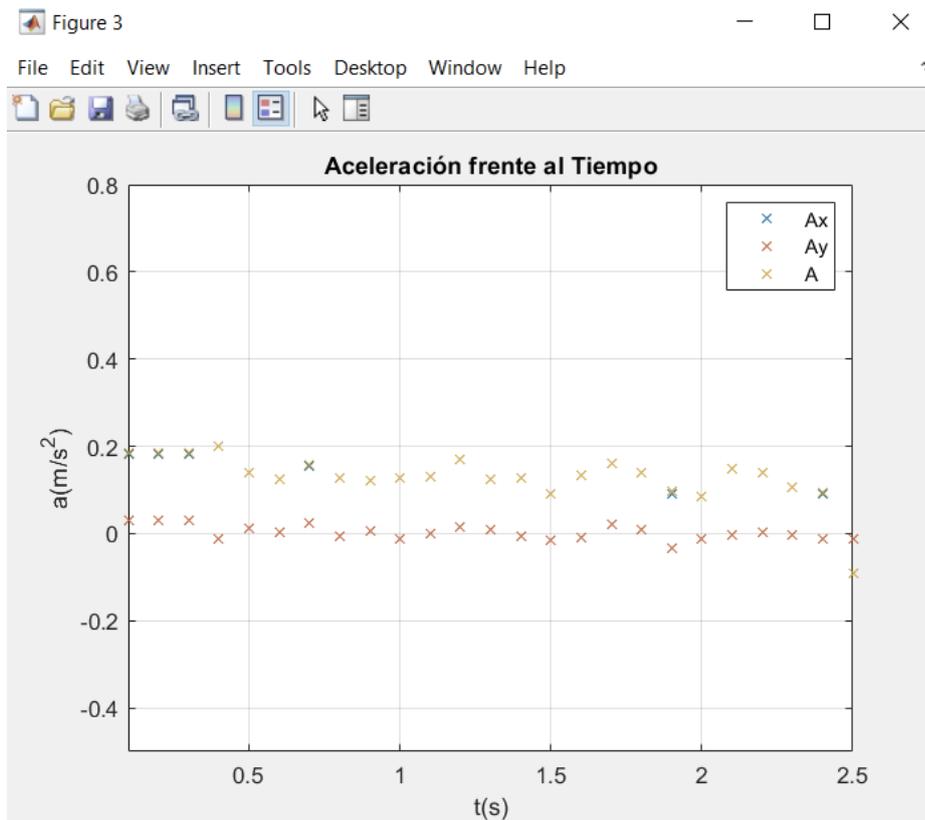


Figura 60. Aceleración frente al tiempo. MRUV. MATLAB.

La fricción producida por fluidos hace que los valores de la aceleración del volante no sean constantes, pero varíen en torno a un valor constante.

Se representan en una gráfica las aceleraciones obtenidas en MATLAB junto con el valor es la pendiente de la velocidad,  $A = 0,1288 \pm 0,0009 \text{ m/s}^2$  representado en la gráfica por la recta horizontal naranja.

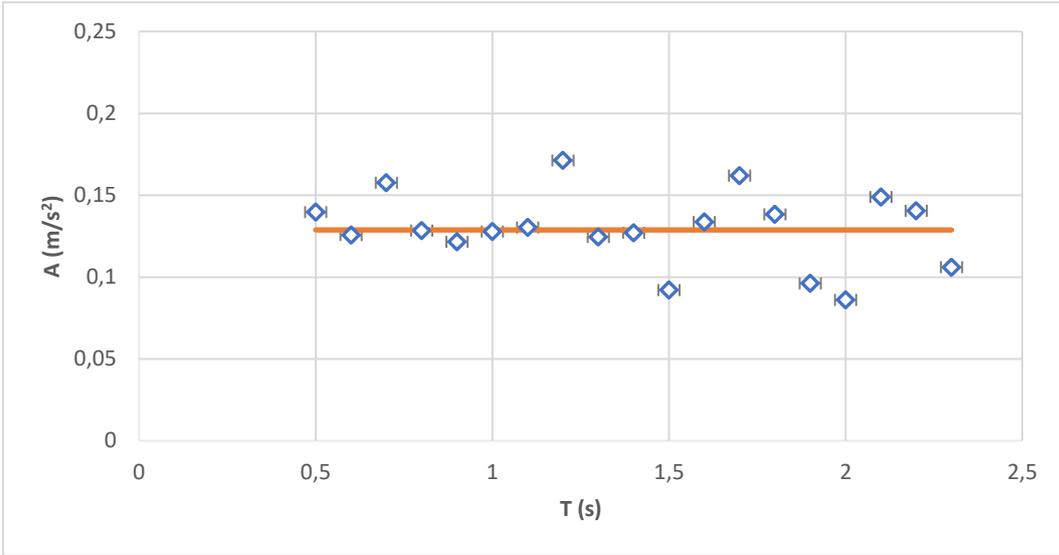
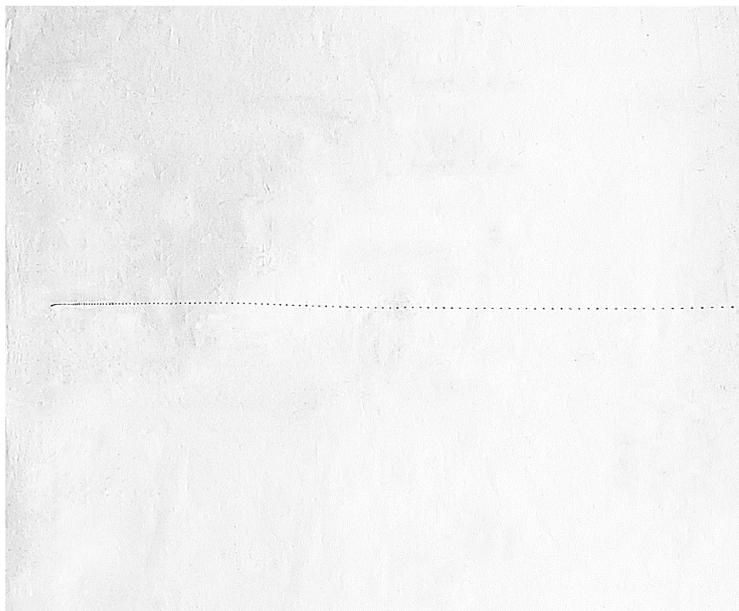


Figura 61. Aceleración frente al tiempo y cte. MRUV. MATLAB.

### RESULTADO EXPERIMENTAL PAPEL ELECTROCONDUCTOR

A continuación se muestran los resultados obtenidos a partir de los impulsos del electrodo central del volante en el papel electroconductor. La frecuencia con la se realizan los impulsos es 50 Hz. Por lo tanto, entre cada posición marcada hay  $1/50$  segundos.



*Figura 62. Papel electroconductor. MRUV*

Igual que en el movimiento rectilíneo se toma el mismo intervalo de tiempo que en Matlab, 0,1 segundos entre pulsaciones para aproximar los datos lo máximo posible. Por lo tanto, se tomarán en cuenta las posiciones a intervalos de 5 en 5. Se medirá entre el inicio de las pulsaciones la distancia entre ellas para así poder hallar la velocidad y la aceleración.

Se puede observar en los datos obtenidos de MATLAB que el movimiento se inicia en el fotograma inicial del video. En consecuencia, las medidas tomadas en el papel electroconductor se iniciarán desde el instante inicial también.

Se sitúan los ejes en el papel electroconductor haciendo coincidir el eje de las X con el movimiento seguido por el volante, y así obtener el movimiento en el eje Y siempre igual a cero.

Se anotan en una tabla los resultados obtenidos, y se muestran en gráficas para poder observarlos de una manera más visual.

T ± 0,02 (s)	S ± 0,001 (m)	V (m/s)	A (m/s <sup>2</sup> )
0,10	0,000	0,03 ± 0,03	0,0 ± 0,5
0,20	0,003	0,03 ± 0,03	0,0 ± 0,5
0,30	0,006	0,03 ± 0,03	0,0 ± 0,5
0,40	0,011	0,05 ± 0,03	0,2 ± 0,6
0,50	0,018	0,07 ± 0,03	0,2 ± 0,7
0,60	0,025	0,07 ± 0,03	0,0 ± 0,7
0,70	0,034	0,09 ± 0,04	0,2 ± 0,8
0,80	0,044	0,10 ± 0,04	0,1 ± 0,8
0,90	0,055	0,11 ± 0,04	0,1 ± 0,8
1,00	0,068	0,13 ± 0,05	0,2 ± 0,9
1,10	0,082	0,14 ± 0,05	0,1 ± 1,0
1,20	0,097	0,15 ± 0,05	0,1 ± 1,0
1,30	0,113	0,16 ± 0,05	0,1 ± 1,0
1,40	0,130	0,17 ± 0,05	0,1 ± 1,1
1,50	0,149	0,19 ± 0,06	0,2 ± 1,2
1,60	0,169	0,20 ± 0,06	0,1 ± 1,2
1,70	0,190	0,21 ± 0,06	0,1 ± 1,2
1,80	0,212	0,22 ± 0,06	0,1 ± 1,3
1,90	0,235	0,23 ± 0,07	0,1 ± 1,3
2,00	0,260	0,25 ± 0,07	0,2 ± 1,4
2,10	0,286	0,26 ± 0,07	0,1 ± 1,4
2,20	0,313	0,27 ± 0,07	0,1 ± 1,5
2,30	0,341	0,28 ± 0,08	0,1 ± 1,5
2,40	0,370	0,29 ± 0,08	0,1 ± 1,6
2,50	0,400	0,30 ± 0,08	0,1 ± 1,6

Tabla 8. Datos del volante en el tiempo. MRUV. Papel electroconductor.

## POSICIÓN

Se realiza el ajuste por mínimos cuadrados de las posiciones obtenidas con el papel electroconductor, y se representan en una gráfica.

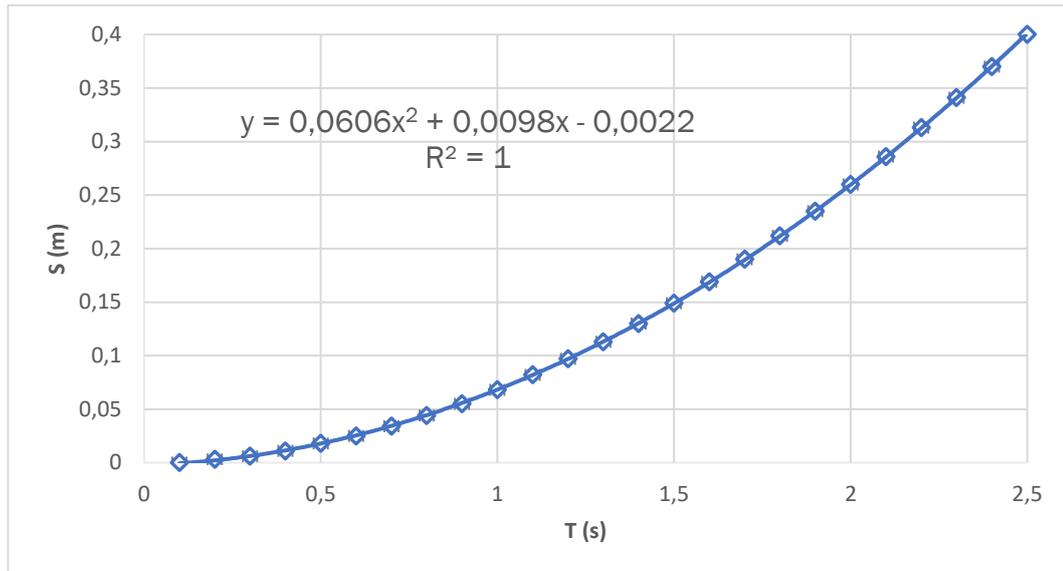


Figura 63. Ajustes mínimos cuadrados. Posición frente al tiempo. MRUV. Papel electroconductor.

## VELOCIDAD

Se realiza el ajuste por mínimos cuadrados de las velocidades obtenidas con el papel electroconductor, y se representan en una gráfica.

- La correlación obtenida es: 0,99905407
- La pendiente obtenida es:  $0,0765 \pm 0,0017$ .
- La ordenada en el origen es:  $0,1185 \pm 0,0010$ .

Por lo tanto, la aceleración será:  $A = 0,1185 \pm 0,0010 \text{ m/s}^2$ .

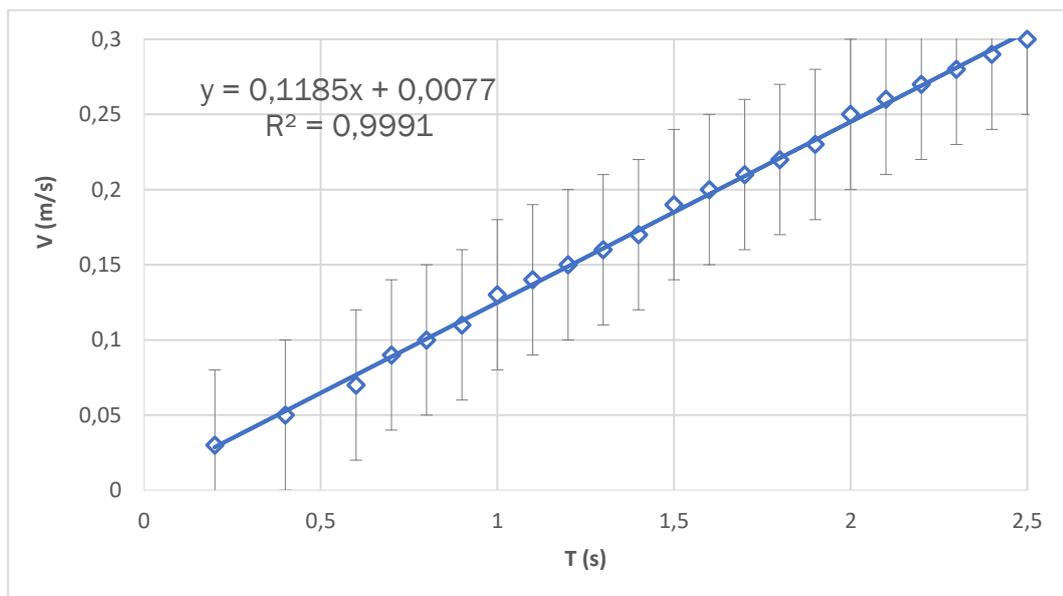


Figura 64. Ajustes mínimos cuadrados. Velocidad frente al tiempo. MRUV. Papel electroconductor.

## ACELERACIÓN

La fricción producida por fluidos hace que los valores de la aceleración obtenida con el papel electroconductor no sean constantes. Pero varían en torno a un valor constante positivo, haciendo que el volante aumente su velocidad.

Se representan en una gráfica los valores que toma la aceleración a lo largo del tiempo junto con el valor de la pendiente de la velocidad,  $A = 0,1185 \pm 0,0010$  m/s<sup>2</sup>, recta horizontal naranja.

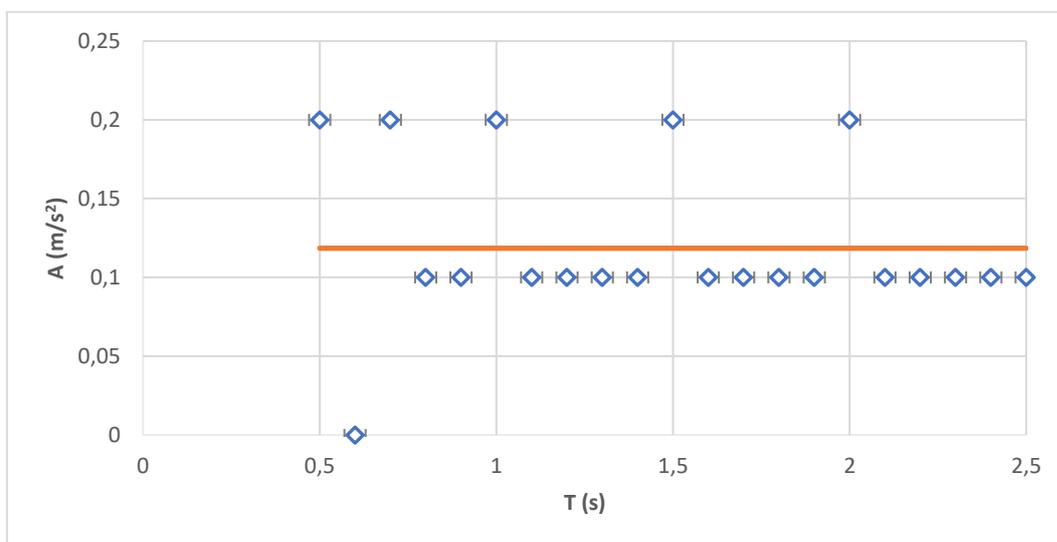


Figura 65. Aceleración frente al tiempo y cte. MRUV. Papel electroconductor.

### COMPARACIÓN

Se va a comparar ambos resultados experimentales junto con el valor teórico, para probar la efectividad del programa.

En primer lugar, se realiza la comprobación teórica.

En ambos experimentos se ha obtenido una aceleración cercana a la teórica  $A = 0,122 \pm 0,007 \text{ m/s}^2$ . Los resultados experimentales tienen ligera diferencia al resultado teórico debido a que las condiciones externas no son ideales, el colchón de aire no es perfecto y existe una pequeña fuerza de rozamiento.

En segundo lugar, se comparan los datos experimentales obtenidos a partir del papel electroconductor y de MATLAB. En estos datos existen diferencias, al igual que en la práctica anterior, por diferentes circunstancias externas:

- El instante inicial de ambos movimientos no es exactamente el mismo, se ha aproximado lo máximo posible, pero puede haber errores de décimas de segundo entre ambos movimientos.
- Los ejes de coordenadas se han aproximado, pero no son exactos, por ello se ha preferido medir la distancia entre puntos que se considera más precisa.
- Pueden existir errores de  $\pm 1 \text{ mm}$  en la medición del papel electroconductor por el instrumento de medición utilizado

#### Interpretación de los resultados:

Se comparan ambos resultados experimentales:

El desplazamiento absoluto observado en MATLAB es de  $0,397 \pm 0,002 \text{ m}$ , muy similar al obtenido por el papel electroconductor  $0,400 \pm 0,001 \text{ m}$ . El desplazamiento a lo largo del tiempo también es similar.

La velocidad máxima observada en MATLAB es de  $0,30 \pm 0,13 \text{ m/s}$ , similar a la obtenida por el papel electroconductor  $0,30 \pm 0,08 \text{ m/s}$ . Las velocidades tomadas por el volante a lo largo del tiempo son similares en ambas.

Las aceleraciones tomadas por el volante a lo largo del tiempo son similares en ambas, varían en torno a unos valores constantes similares. Aunque existen algunas diferencias entre ellas:

- Por una parte, las aceleraciones observadas en MATLAB varían próximas a  $0,1288 \pm 0,0009 \text{ m/s}^2$  con subidas y bajadas constantes en torno a ese valor, haciendo que disminuya la velocidad del volante de manera progresiva.

- Por la otra parte, las aceleraciones observadas en el papel electroconductor tienen cambios más bruscos, debido a la baja precisión del instrumento de medida, y a trabajar a bajas velocidades. En este caso las aceleraciones observadas en el papel electroconductor varían próximas a  $0,1185 \pm 0,0010 \text{ m/s}^2$

Al comparar ambos resultados se observa un resultado satisfactorio. Hay pequeñas diferencias en ambas prácticas, pero son asumibles debido a la inexactitud por las circunstancias externas mencionadas anteriormente.

## CAPÍTULO 6. CONCLUSIONES

### 6.1. CONCLUSIONES GENERALES

En el transcurso de este proyecto se han alcanzado diversas conclusiones significativas. Partiendo de los objetivos previamente fijados:

- La implementación de la metodología propuesta para el seguimiento del volante ha demostrado ser efectiva en condiciones simuladas. Los resultados obtenidos respaldan la viabilidad de la visión artificial para la detección del volante.
- Se ha desarrollado un algoritmo en una función externa que detecta con éxito el volante en cada fotograma.
- Se ha realizado el estado del arte del seguimiento de objetos.
- Se ha descrito la instrumentación necesaria, y el funcionamiento de la mesa neumática.
- Se han obtenido las gráficas de posición, velocidad y aceleración de cada movimiento.
- Se ha conseguido que la programación sea aplicable independientemente de la calidad de la cámara, gracias a unos argumentos de entrada solicitados al usuario al inicio del programa.
- Se ha conseguido que las condiciones de iluminación afecten los menos posible a los resultados obtenidos, al seleccionar la detección de objetos por color con los umbrales lo más restrictivos posibles. La altura de grabación finalmente no ha supuesto ningún inconveniente ya que la equivalencia entre pixeles y cm lo regula automáticamente en función de esta altura.
- Se han realizado pruebas satisfactorias del sistema de seguimiento de objetos, comparando los resultados de MATLAB con los obtenidos en el papel electroconductor, y los teóricos.

La codificación desarrollada es versátil y puede ajustarse a distintas frecuencias en la representación gráfica de las posiciones deseadas.

En conclusión, se ha implementado un programa que permite realizar el seguimiento del volante a lo largo del tiempo satisfactoriamente. Abriendo un abanico de posibilidades para el desarrollo de la visión artificial en el ámbito educativo.

## 6.2. LÍNEAS FUTURAS

Este trabajo sienta las bases para investigaciones futuras que podrían abordar la optimización del rendimiento y la implementación de otro tipo de prácticas con diferentes movimientos.

Algunas líneas para investigaciones futuras:

- Optimizar el programa.
- Realizar otros movimientos que involucren varios volantes.
- Instalar complementos de MATLAB para grabar con la webcam el movimiento directamente, sin necesidad de la cámara del móvil.

## BIBLIOGRAFÍA

En el transcurso de este proyecto se han consultado referencias bibliográficas para conocimientos teóricos, el estudio del arte de la visión artificial, o para el uso de determinadas funciones de MATLAB.

- (1). *Apuntes Física I.* (2018). Ingeniería en Tecnologías Industriales. Universidad de Valladolid.
- (2). *Escribir datos en hojas de cálculo de Excel.* (s.f.). Obtenido de Centro de ayuda. MATLAB: [https://es.mathworks.com/help/matlab/import\\_export/exporting-to-excel-spreadsheets.html](https://es.mathworks.com/help/matlab/import_export/exporting-to-excel-spreadsheets.html)
- (3). Fernández, J. L. (s.f.). *Movimiento Rectilíneo Uniformemente Acelerado.* Obtenido de FISICALAB: <https://www.fisicalab.com/apartado/mrua>
- (4). *Identificación de colores con Matlab.* (2015). Obtenido de Youtube: <https://www.youtube.com/watch?v=QJPu7R1K3CU>
- (5). López, E. d., & Trespaderne, F. M. (2012). *Visión artificial industrial. Procesamiento de imágenes para inspección automática y robótica.* Universidad de Valladolid, Secretariado de Publicaciones.
- (6). López, E. d., & Trespaderne, F. M. (6 de noviembre de 2023). *Visión artificial industrial. Procesamiento de imágenes para inspección automática y robótica.* Obtenido de Universidad de Valladolid: <http://www.librovision.eii.uva.es/index.html>
- (7). *Procesamiento de imágenes y Visión artificial.* (s.f.). Obtenido de MATLAB: <https://es.mathworks.com/solutions/image-video-processing.html>
- (8). *Reconocimiento de patrones.* (s.f.). Obtenido de MATLAB: <https://es.mathworks.com/discovery/pattern-recognition.html>
- (9). *Segmentación por colores utilizando el espacio de color.* (s.f.). Obtenido de Centro de ayuda. MATLAB: <https://es.mathworks.com/help/images/color-based-segmentation-using-the-l-a-b-color-space.html>
- (10). *Segmentación y análisis de imágenes.* (s.f.). Obtenido de Centro de ayuda. MATLAB: <https://es.mathworks.com/help/images/image-analysis.html>

(11). *Técnicas de visión artificial y procesamiento de imágenes con MATLAB*. (2022). Obtenido de Youtube: [https://www.youtube.com/watch?v=sJ2eAW\\_Xg9M](https://www.youtube.com/watch?v=sJ2eAW_Xg9M)