

UNIVERSIDAD DEL VALLE DE GUATEMALA  
Facultad de Ingeniería



**Diseño de Sistema Autónomo de Seguimiento de Atletas en la  
Natación por Medio de una Cámara de Video**

Trabajo de graduación presentado por José Gerardo Molina Rodríguez  
para optar al grado académico de Licenciado en Ingeniería Mecatrónica

Guatemala,

2019





UNIVERSIDAD DEL VALLE DE GUATEMALA  
Facultad de Ingeniería



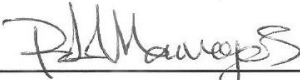
**Diseño de Sistema Autónomo de Seguimiento de Atletas en la  
Natación por Medio de una Cámara de Video**

Trabajo de graduación presentado por José Gerardo Molina Rodríguez  
para optar al grado académico de Licenciado en Ingeniería Mecatrónica

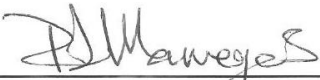
Guatemala,

2019

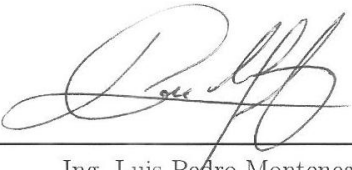
Vo.Bo.:

(f)   
Ing. Pablo Mazariegos

Tribunal Examinador:

(f)   
Ing. Pablo Mazariegos

(f)   
MSc. Carlos Esquit

(f)   
Ing. Luis Podro Montenegro

Fecha de aprobación: Guatemala, 3 de diciembre de 2019.

Muchas veces pensamos que el camino es difícil y que solo al llegar al destino seremos felices, pero yo puedo asegurar que he disfrutado este camino, que, aunque ha sido difícil, me ha enseñado a vivir, a aprender y a valorar a las personas que me rodean. Este proyecto es mucho más que un trabajo de graduación es la recopilación de todo lo que he aprendido en este camino que seguiré recorriendo por el resto de mi vida, no veo esto como el final, sino que al contrario lo veo como el inicio y la puerta a muchas otras oportunidades.

Dejo este escrito filosófico al inicio de este trabajo, para que al momento de leer este trabajo se pueda entender un poco mejor mi forma de pensar y de esta manera se pueda entender el por qué de algunas de las cosas que aquí plasmo. Invito al lector a tomar una tasa de café, unas champurradas y a disfrutar de la parte de mi camino que dejo en este trabajo, en donde describo el proceso que seguí para desarrollar el diseño de un prototipo que espero pueda apoyar a futuros atletas en Guatemala.

Definitivamente no voy a dejar pasar la oportunidad de agradecer a todas las personas que me han apoyado en esta parte del camino, porque realmente son importantes en mi vida y espero que lo sigan siendo en el futuro. No me gusta nombrar y numerar específicamente a personas porque son muchas (y es posible que se me pueda escapar alguna) y si estas leyendo esto quiero que sepas que te estoy muy agradecido. Sin más que decir, dedico este trabajo a ti, que lo estas leyendo porque de una u otra forma te estas encontrando con mi camino, espero que sin importar la razón por la que estas leyendo este trabajo, lo disfrutes tanto como yo lo hice en su momento.

<b>Prefacio</b>	III
<b>Lista de figuras</b>	VIII
<b>Lista de cuadros</b>	IX
<b>Resumen</b>	XI
<b>1. Introducción</b>	1
<b>2. Antecedentes</b>	2
2.1. Sistema de seguimiento de jugadores por medio de Visión de Computadora . . .	2
2.2. Registro de entrenamientos basado en video . . . . .	3
2.3. Efectividad de software en el seguimiento de movimiento debajo del agua . . .	4
<b>3. Justificación</b>	6
<b>4. Objetivos</b>	8
4.1. Objetivo general . . . . .	8
4.2. Objetivos específicos . . . . .	8
<b>5. Alcance</b>	9
<b>6. Marco teórico</b>	10
6.1. La natación . . . . .	10
6.1.1. Estilo Crol . . . . .	10
6.1.2. La Braza de Pecho . . . . .	11
6.1.3. Nado de espalda . . . . .	11
6.1.4. ¿Qué es necesario medir en la natación? . . . . .	12
6.2. La natación, un deporte olímpico . . . . .	14
6.2.1. Récords de la natación en los Juegos Olímpicos . . . . .	14
6.3. Visión de Computadora . . . . .	15
6.4. PyQt . . . . .	15
6.5. OpenCV . . . . .	16

6.6. Procesamiento de imágenes	16
6.7. Filtrado de imágenes	17
6.8. Imágenes digitales	19
6.9. Thresholding y extracción de información	20
6.10. Visual Servoing	20
6.10.1. Position Based Visual Servoing (PBVS)	21
6.10.2. Image Based Visual Servoing (IBVS)	21
6.11. Raspberry Pi 3	21
<b>7. Diseño mecánico del sistema</b>	<b>23</b>
7.1. Requisitos del sistema	23
7.2. Primer prototipo	24
7.2.1. Diseño del prototipo	24
7.2.2. Construcción del prototipo	24
7.2.3. Pruebas de validación del prototipo	25
7.3. Segundo prototipo	27
7.3.1. Diseño del prototipo	27
7.3.2. Construcción del prototipo	28
7.3.3. Pruebas de validación del prototipo	29
7.4. Tercer prototipo	30
7.4.1. Diseño general del prototipo	30
7.4.2. Diseño de llantas	31
7.4.3. Desplazamiento horizontal	32
7.4.4. Diseño de sujeción de cámara	33
7.4.5. Bases para sujeción de cable	33
7.4.6. Construcción del prototipo	33
7.4.7. Pruebas de validación del prototipo	34
<b>8. Sistema de seguimiento por medio de una cámara de video</b>	<b>39</b>
8.1. Requisitos del sistema	39
8.2. Diseño de programa de seguimiento	39
8.3. Sistema de seguimiento utilizando la Raspberry Pi 3	40
8.3.1. Pruebas de validación utilizando la Raspberry Pi 3	41
8.4. Sistema de seguimiento utilizando una computadora	41
8.4.1. Pruebas de validación utilizando una computadora	41
<b>9. Implementación de motor</b>	<b>43</b>
9.1. Requisitos del sistema	43
9.2. Diseño de sistema	43
9.3. Pruebas de validación	45
<b>10. Selección de motor</b>	<b>46</b>
<b>11. Interfaz gráfica</b>	<b>50</b>
11.1. Requisitos de la interfaz gráfica	50
11.2. Diseño de la interfaz gráfica	50
11.3. Pruebas de validación	52

<b>12. Materiales adicionales</b>	<b>55</b>
12.1. Requisitos de los materiales . . . . .	55
12.2. Cable de acero con alma de yute . . . . .	55
12.3. Perfiles de aluminio . . . . .	56
12.3.1. Resistencia al agua y a la humedad . . . . .	57
12.4. Cable resistente al agua . . . . .	57
<b>13. Conclusiones</b>	<b>59</b>
<b>14. Recomendaciones</b>	<b>60</b>
<b>15. Bibliografía</b>	<b>61</b>
<b>16. Anexos</b>	<b>63</b>
16.1. ANEXO A . . . . .	63
16.2. ANEXO B . . . . .	63
16.3. ANEXO C . . . . .	64
16.3.1. ANEXO C.1 . . . . .	64
16.3.2. ANEXO C.2 . . . . .	65
16.4. ANEXO D . . . . .	66
16.4.1. ANEXO D.1 . . . . .	67
16.4.2. ANEXO D.2 . . . . .	68
16.4.3. ANEXO D.3 . . . . .	69
16.4.4. ANEXO D.4 . . . . .	70
16.4.5. ANEXO D.5 . . . . .	71
16.4.6. ANEXO D.6 . . . . .	72

---

## Lista de figuras

---

1. Visualización de seguimiento en proyecto de Universidad de Ámsterdam. . . . .	3
2. Distintos tipos de marcas utilizados en el experimento. . . . .	4
3. Resultados presentados en artículo de Journal of Sports and Medicine. . . . .	5
4. Alineación horizontal con poca resistencia al agua. . . . .	11
5. Ejecución del brazo en el estilo Crol. . . . .	11
6. Acción y coordinación de piernas en el estilo Crol. . . . .	12
7. Movimiento de arrastre en estilo Braza de Pecho. . . . .	12
8. Movimiento de pataleo en estilo Braza de Pecho. . . . .	13
9. Movimiento de cuerpo en nado de espalda. . . . .	13
10. Estructura básica de OpenCV. . . . .	17
11. Ejemplo del Box Blur . . . . .	18
12. Ejemplo de filtro Gaussiano . . . . .	19
13. Proyección de una imagen. . . . .	19
14. Diferencia entre el PBVS y el IBVS. . . . .	20
15. Sistema controlado por un PBVS. . . . .	21
16. Sistema controlado por IBVS. . . . .	22
17. Prototipo 1: Slider utilizado como inspiración. . . . .	24
18. Prototipo 1: Modelo 3D . . . . .	25
19. Prototipo 1: Base del sistema . . . . .	26
20. Prototipo 1: Rieles de madera . . . . .	26
21. Prototipo 1: Carrito 1. . . . .	27
22. Prototipo 2: Sistema de cojinetes . . . . .	28
23. Prototipo 2: Modelo 3D . . . . .	28
24. Prototipo 2: Mecanismo de rodamientos MDF . . . . .	29
25. Prototipo 2: Fallo mecanismo de rodamientos . . . . .	29
26. Prototipo 2: Mecanismo de rodamientos PLA . . . . .	30
27. Prototipo 3: Modelo 3D . . . . .	31
28. Prototipo 3: Llanta MDF . . . . .	32
29. Prototipo 3: Llantas . . . . .	33
30. Prototipo 3: Comparación de llantas . . . . .	34
31. Prototipo 3: Vistas de prototipo final . . . . .	35

32. Prototipo 3: Piñón eje motor	36
33. Prototipo 3: Mecanismo cojinete	36
34. Prototipo 3: Sujeción de cámara	37
35. Prototipo 3: Carrito construido	37
36. Prototipo 3: Caja de cemento	37
37. Prototipo 3: Funcionamiento	38
38. Sistema de seguimiento: Conexión RBPi 3	40
39. Sistema de seguimiento: Conexión computadora	42
40. Sistema de seguimiento: Prueba de seguimiento.	42
41. Implementación de motor: Prueba de seguimiento.	44
42. Interfaz de Usuario: Mapa de la aplicación	51
43. Interfaz de Usuario: Diseño	52
44. Interfaz de Usuario: Resultado final	53
45. Interfaz de Usuario: Prueba con prototipo	53
46. Interfaz de Usuario: Prueba con prototipo desde cámara de video.	54
47. Interfaz de Usuario: Generar videos.	54
48. Interfaz de Usuario: Abrir videos.	54
49. Materiales adicionales: Cable de acero	56
50. Circuito para conexión de motores	63
51. Diagrama de flujo de programa en Arduino	64
52. Circuito para conexión de motores	65
53. Plano del sistema completo Hoja 1	67
54. Plano del sistema completo Hoja 2	68
55. Plano del sistema completo Hoja 3	69
56. Plano del sistema completo Hoja 4	70
57. Plano del sistema completo Hoja 5	71
58. Plano del sistema completo Hoja 6	72

1. Competencias de natación en las olimpiadas	14
2. Records de natación actuales	15
3. Características técnicas de la Raspberry Pi 3	22
4. Materiales para prototipo 1, sistema de rieles.	25
5. Materiales para prototipo 1, carrito.	26
6. Implementación de motor: Caracteres enviados.	44
7. Pruebas de velocidad motor 3V	47
8. Selección de motor: Pruebas de Velocidad 5V	47
9. Selección de motor (Radio = 0.008)	47
10. Selección de motor (Radio = 0.02)	48
11. Selección de motor (Radio = 0.05)	48
12. Motores evaluados	48
13. Motores evaluados, enlaces a página y hoja de datos.	49

En la actualidad la tecnología se ha convertido en un aspecto muy importante para los deportistas de alto rendimiento, esto debido a que está al alcance de todos y permite obtener información sobre el rendimiento, aspectos técnicos, tácticos, etc. En el siguiente trabajo se ha desarrollado el diseño de un sistema que tiene como objetivo principal dar seguimiento a un nadador por medio de una cámara de video para el monitoreo de sus prácticas y sesiones. Para esto se diseñó el prototipo de un carrito que se puede desplazar horizontalmente a la orilla de una piscina siguiendo a un nadador, tomando como un riel guía un cable que se encuentra tensado y sujetado en ambos extremos del espacio en que se desea desplazar y por medio de una cámara de video, que se encuentra montada en él, puede detectar al nadador para mantenerlo dentro el cuadro de la misma y grabar su recorrido.

El carrito fue diseñado utilizando técnicas y software de modelado en 3D y se maquinó utilizando herramientas como una cortadora láser e impresoras 3D, se utilizó como riel un cable de acero tensado y como sensor una cámara web con conexión USB. Para realizar el seguimiento se utilizó OpenCV programado en Python, para que por medio de técnicas de visión de computadora se pudiera detectar un objeto de un color específico y se pudiera determinar la dirección en que este se desplaza. Se diseñó una interfaz gráfica, que también fue programada utilizando el lenguaje de programación Python y herramientas de las librerías de Qt, que permite iniciar rutinas con el sistema desde una computadora y que además brinda una retroalimentación de video inmediato y almacena en una colección de videos, todas las grabaciones de cada una de las sesiones en las que se ha utilizado el dispositivo. Adicional a esto se realizó una investigación para proporcionar materiales que pueden ser utilizados para la fabricación de un prototipo que podría ser expuesto a salpicaduras de agua, para el caso en el que se desee construir dicho prototipo para ser expuesto a salpicaduras de agua y humedad constante, también se calculó la velocidad a la que es necesario que el dispositivo se desplace para poder dar seguimiento a atletas de alto rendimiento y en base a esto se eligió un listado de motores que pueden ser utilizados para alcanzar una velocidad similar y mantenerlos dentro del encuadre de la cámara. Por último, se escribió un manual en el cual se describen los pasos detallados que se deben de seguir para poder ensamblar y utilizar el prototipo mecánico y cómo enlazar a este la aplicación de usuario para poder replicar el funcionamiento del mismo.

Se realizó pruebas de funcionamiento con el prototipo y se comprobó su correcto funcionamiento detectando objetos de diferentes colores, grabando videos y también se determinó que a la velocidad a la que este se desplaza no es suficiente para dar seguimiento a atletas de alto rendimiento, por lo que se procedió a investigar motores en el mercado que pudieran proporcionar la velocidad necesaria, para esto se realizó un listado que incluyen un enlace al vendedor y a su hoja de datos.

A nivel mundial la tecnología está abarcando muchos campos, incluyendo el deporte, en el cual se está tratando a los atletas para mejorar su rendimiento y condición física. En Guatemala existen muchas iniciativas que están proporcionando apoyo en esta área para impulsar instituciones como lo son el Comité Olímpico Guatemalteco (COG por sus siglas en español) y la Confederación Deportiva Autónoma de Guatemala (CDAG por sus siglas en español). En el país el deporte no ha tenido un desempeño impactante como a muchas personas les gustaría y aún así existen personas y entidades que están dispuestas a trabajar por un mejor futuro para las distintas ramas del deporte que representan a nuestro país de forma internacional. Para el correcto desarrollo de los atletas suele utilizarse el conocimiento de múltiples entrenadores y expertos que brindan el grano de arena que permite que los atletas de alto rendimiento puedan alcanzar grandes metas. Hoy en día es posible crecer como atleta en las distintas disciplinas por medio de los métodos convencionales y obteniendo ayuda de la tecnología para mejorar los resultados.

En el proyecto descrito en el siguiente texto se desarrolló el diseño de un sistema que tiene como objetivo principal dar seguimiento a un nadador mientras se desempeña en sus actividades de este deporte, esto dando la oportunidad de observar durante su recorrido en la piscina. A continuación, se mostrará una investigación que brinda información necesaria para entender algunos temas importantes a conocer sobre dicha disciplina, también se muestra un reporte de los pasos seguidos para el diseño del sistema descrito anteriormente y los resultados obtenidos. El sistema de seguimiento de un nadador por medio de una cámara de video cuenta con un sistema mecánico, que consta de un carro que se desplaza horizontalmente en el cual se puede montar una o dos cámaras para brindar seguimiento al atleta, un sistema de software que permite ubicar y seguir a un nadador que esté utilizando una gorra de un color específico y por medio de este controlar un sistema motorizado que permita desplazar el sistema mecánico. El prototipo desarrollado también cuenta con una interfaz gráfica que permite ver un feed de video, videos almacenados anteriormente y controlar el sistema completo.

Seguir a los atletas mientras practican un deporte para obtener información útil sobre su rendimiento puede ser una tarea muy codiciada, pero también difícil de realizar. En la actualidad con el fácil acceso que se tiene a cámaras de buena calidad y el poder computacional que puede procesar la información obtenida, este tema se ha abierto a nuevas oportunidades. A continuación, se presentan algunos proyectos que fueron revisados como parte de investigación sobre algunos sistemas ya existentes o en proceso de implementación para tener una idea de lo que se está trabajando en el mundo en este campo de investigación.

### 2.1. Sistema de seguimiento de jugadores por medio de Visión de Computadora

En la Universidad de Ljubljana en Eslovenia en el año 2000 se publicó un artículo sobre un sistema de seguimiento para atletas en disciplinas de salón, osea que se realizan en ambientes cerrados. En este trabajo presenta resultados de un sistema en el cual tomaron imágenes y corrigieron la calibración del lente para poder tener una visión más amplia y plana del campo. Posteriormente procedieron a capturar imágenes y a tratar de realizar un seguimiento de los jugadores en el campo de visión, para esto utilizaron varias metodologías.

1

- La primera metodología que utilizaron fue por medio de sustracción de imágenes, se tomó una imagen del campo sin jugadores y se le restó la imagen con los jugadores en ella por lo que de esta forma quedarían solamente en la imagen resultante los jugadores. Posteriormente realizaron un threshold a la imagen resultante y obtuvieron algunos puntos que efectivamente eran los jugadores. El problema es que en la imagen luego del procesamiento se encontraron otros puntos que aparecían debido a sombras o ruido de la imagen por lo que su extracción de información no era tan certera se esperaba.

- La segunda metodología que utilizaron fue seguimiento por medio de detección de los píxeles representativos de los jugadores. Esto lo realizaron extrayendo las formas más básicas del jugador, de aproximadamente 10-15 píxeles y se hizo una búsqueda separando los canales de cada color. Con esta metodología obtuvieron mejores resultados que con la anteriormente mencionada.
- En esta última metodología que se presenta en el trabajo utilizaron un seguimiento por medio del color, en donde utilizaron una medida de similitud calculada con una expresión matemática que relaciona la imagen con el color buscado.

Como conclusión a este proyecto se dijo que lograron realizar una detección de los jugadores en un entorno cerrado por medio de tres metodologías, dejando abierto a qué se podría realizar posteriormente con esta información obtenida.

## 2.2. Registro de entrenamientos basado en video

En 2007 la Universidad de Ámsterdam y un grupo de Intelligent Systems Lab en Holanda realizaron un artículo científico en donde describen la aplicación del registro de sesiones de entrenamiento basados en video, utilizando como argumento principal que por lo general los entrenadores no tienen suficientes medios para justificar sus entrenamientos. En este proyecto se obtuvo información de los nadadores siguiéndolos con múltiples cámaras durante su sesión de entrenamiento.

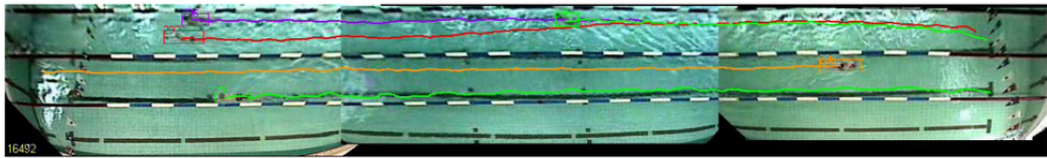


Figura 1: Visualización de seguimiento en proyecto de Universidad de Ámsterdam.

2

Para realizar la etapa de experimentación colocaron 3 cámaras en trípodes altos a lo largo de una piscina de 50 metros para poder obtener una imagen aérea de los nadadores, como se muestra en la Figura 1. Posteriormente se tomó 45 minutos de video que fue procesado en computadora para tener un campo de visión adecuado. Posteriormente por medio de métodos de visión de computadora lograron seguir a los nadadores en distintos carriles de la piscina con algunos inconvenientes. El seguimiento de los atletas no fue totalmente autónomo ya que debieron arreglar algunos puntos manualmente porque el procesamiento confundió algunos de los nadadores en distintos carriles. 2

### 2.3. Efectividad de software en el seguimiento de movimiento debajo del agua

Hace algunos años, en el 2013, la revista *Journal of Sports and Medicine* publicó un artículo en el que se evalúa el funcionamiento de un software especializado en el seguimiento de ciertas marcas realizadas en el cuerpo de una persona mientras está sumergida debajo del agua. El principal argumento de este experimento fue comprobar el funcionamiento de dicho software ya que argumentan que el poder obtener información cinemática tanto de atletas permite evaluar aspectos técnicos durante competiciones de alto rendimiento.

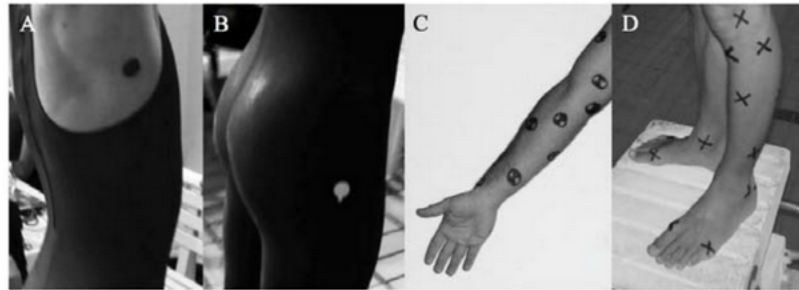


Figura 2: Distintos tipos de marcas utilizados en el experimento.

3

Para obtener la información que deseaban se utilizó un programa llamado COM y DVP, los cuales se pueden conseguir comercialmente, en estos desarrollaron un algoritmo que permite obtener la posición de las marcas realizadas en la anatomía de los atletas. 2 Luego de haber diseñado el algoritmo este fue puesto a prueba en distintas situaciones, se tomó video y se realizó el seguimiento manual y automático. Luego de analizar los datos pudieron observar por medio de gráficos de los datos obtenidos con los distintos métodos una representación del movimiento de los atletas que estaban siguiendo. 3 3

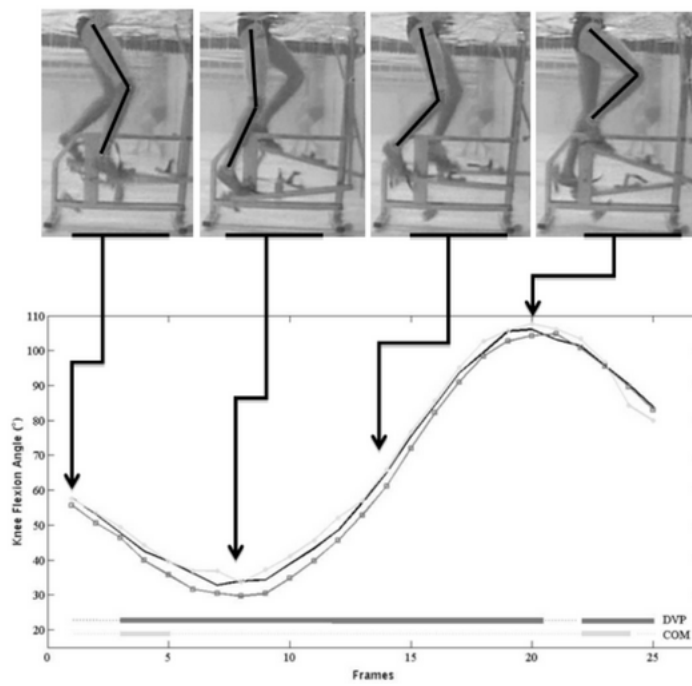


Figura 3: Resultados presentados en artículo de Journal of Sports and Medicine.

3

Hoy en día la exigencia en el deporte de alto rendimiento ha llegado al punto de demandar tecnología para mejorar el rendimiento de los atletas al máximo. En Guatemala hay muchas disciplinas que han dado frutos en competencias a nivel internacional y por eso entidades como el Comité Olímpico de Guatemala y la Confederación Deportiva Autónoma de Guatemala consideran necesaria la inversión de recursos a estos atletas que quieren mejorar y están dispuestos a adaptarse a nuevas herramientas para sobresalir.

La Universidad del Valle de Guatemala, en su compromiso para apoyar el desarrollo del deporte nacional y por medio de la Facultad de Ingeniería y el incentivo del departamento de Ingeniería Mecatrónica diseñará, con las herramientas y conocimientos adquiridos durante el grado de Licenciatura en Ingeniería Mecatrónica, un sistema que sea de utilidad para los atletas que se desempeñan en la disciplina de natación. Este sistema les permitirá tener un retroalimentación de video de su actuación durante sus entrenamientos y competición para que por medio de este puedan mejorar su técnica y desempeño. La implementación de un sistema como este abre las puertas a mejorar el nivel de los atletas del país proporcionando un entrenamiento más personalizado ya que esto permite introducir otros tipos de tecnología que dependan de imágenes como las que se desean obtener, como por ejemplo un análisis de señales o imágenes que permita detectar fallas en la técnica de los atletas para poder corregirla de forma inmediata y de esta forma mejorar el rendimiento y evitar cualquier tipo de lesión. Este tipo de tecnología podría ser adaptado a otras disciplinas que lo requieran por lo que su alcance no es solamente en la natación.

El sistema de seguimiento de atletas por medio de una cámara de forma autónoma surge de la dificultad de poder proporcionar un seguimiento efectivo, sobre todo en entornos difíciles como lo puede ser una piscina en donde el operario no pueda dar seguimiento al atleta. Por eso se diseñará un sistema que sea capaz de dar seguimiento a un nadador manteniéndolo en el campo de visión de una cámara de video para brindar un video retroalimentación al entrenador y al mismo atleta.

En la actualidad el desarrollo y el crecimiento exponencial en la tecnología de video y el fácil acceso que se tiene a este equipo expanden las posibilidades de desarrollar sistemas

que no requieren mucha complejidad para facilitar tareas que pueden ser complicadas sin ayuda de dicha tecnología y que hace un tiempo era imposible de conseguir. La tarea de seguir a un nadador requiere que el equipo esté protegido para evitar salpicaduras de agua puedan dañarlo y también es necesario que este no afecte o presente ningún riesgo en el desempeño de dichos atletas. Por eso se planea diseñar un sistema que además de cumplir con el cometido de poder proporcionar un seguimiento a los atletas, también sea adecuado a las condiciones que este requiera para evitar inconvenientes en el desarrollo de sus actividades. También es importante para el correcto manejo del hardware, así como para poder observar el retroalimentación de video una aplicación que proporcione la vista del video siguiendo al atleta, así como con otras opciones que le permitan ver videos guardados de entrenamientos anteriores.

### 4.1. Objetivo general

Diseñar un sistema autónomo con la capacidad de dar seguimiento a un nadador por medio de una cámara de video para el monitoreo de la técnica de los atletas.

### 4.2. Objetivos específicos

- Diseñar un sistema mecánico que se desplace horizontalmente en el que se pueda montar una cámara para poder dar seguimiento al atleta.
- Determinar los materiales adecuados para que el sistema pueda soportar salpicaduras de agua debido al entorno en que se encontrará.
- Seleccionar una manera adecuada de envío de datos hacia la aplicación para poder obtener un feedback lo más cercano a tiempo real posible.
- Diseñar una interfaz de usuario que permita ver el video feedback y videos almacenados.
- Desarrollar un manual de usuario sobre el correcto uso y configuración del sistema y la aplicación.

Este proyecto se enfoca en el diseño de un primer prototipo que permita abrir las puertas para apoyar, por medio de la tecnología, a la disciplina de la natación en Guatemala. El prototipo planteado cuenta con un sistema mecánico, un sistema de seguimiento y una interfaz gráfica de usuario.

El sistema mecánico es capaz de desplazarse horizontalmente y permite que en el se monte una cámara de video, como sensor para el sistema de seguimiento y para generar una retroalimentación de video para el atleta. El sistema de seguimiento toma la información de la cámara y genera una señal de control que maneja la dirección en que los motores giran para desplazar el sistema mecánico. La interfaz de usuario permite utilizar el sistema, reproducir videos grabados anteriormente, grabar videos durante los entrenamientos, ver la video retroalimentación y leer un manual de usuario.

Todo este sistema en conjunto provee el diseño de un primer prototipo de un sistema autónomo con la capacidad de dar seguimiento a un objeto (de un color determinado) por medio de una cámara de video.

## 6.1. La natación

Hoy en día la natación, como cualquier otro deporte, se ha sumergido en el mundo de la tecnología y la digitalización. Estos nuevos sistemas permiten obtener una gran cantidad de datos, que posteriormente son procesados con el fin de obtener información que pueda beneficiar a los atletas que se desempeñan en este deporte. Es muy común ver en la televisión gran cantidad de estadísticas que se muestran para realizar comparaciones en las competencias, algunos miden velocidades, tiempos de reacción, tiempos de prolongación, etc. Todo esto permite comparar distintos resultados que se obtienen de mediciones de sensores o aproximaciones de otros eventos. Antes de poder hablar sobre las mediciones realizadas en la natación es importante hablar un poco acerca de la disciplina en sí, como por ejemplo algunos conceptos básicos que deben saber atletas, entrenadores o cualquier persona interesada en este deporte.

En la natación la posición del cuerpo es muy importante puesto que tiene que ser lo más hidrodinámica posible, asegurando que el agua ejerza la menor resistencia posible y tratando de evitar cualquier tipo de fatiga, por esta razón una buena técnica es necesaria para un buen desempeño. Uno de los principales aspectos a verificar para evitar una resistencia grande al agua es la alineación horizontal del cuerpo [4](#), ya que al realizar el movimiento de los brazos y piernas el cuerpo suele moverse para el lado contrario, por lo que se debe intentar que la alineación del cuerpo sea la adecuada para evitar perder velocidad.

### 6.1.1. Estilo Crol

El estilo Crol, es uno de los más conocidos y el más utilizado en las competencias de nado libre, la razón de esto es que es un estilo de natación muy eficaz por la velocidad que se puede llegar a desarrollar y la simplicidad que este refleja. Sin embargo, es muy importante, como ya se mencionó, la técnica correcta para poder abordar de la mejor forma este estilo.

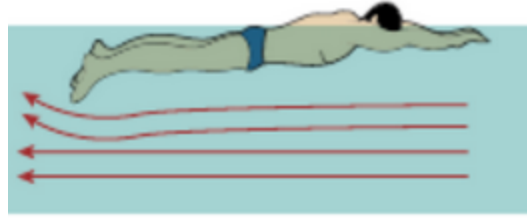


Figura 4: Alineación horizontal con poca resistencia al agua.

En la ejecución los brazos deben seguir ciertos criterios dependiendo de la posición en la que se encuentren.<sup>5</sup> De la misma manera en que el movimiento de los brazos es importante, también es muy importante la acción y coordinación del movimiento de las piernas en el pataleo, asegurando que con una buena técnica se generará una menor resistencia y por ende una mayor velocidad.<sup>4 6</sup>

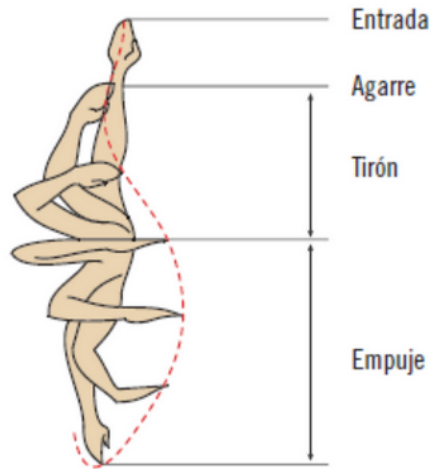


Figura 5: Ejecución del brazo en el estilo Crol.

### 6.1.2. La Braza de Pecho

La Braza de Pecho es otro estilo de la natación con el cual se compite a nivel olímpico. Este puede llegar a ser un estilo cómodo de practicar, pero en comparación con el Crol este es muy lento. En este estilo de nado el cuerpo se debe extender y luego flexionar completamente. En este caso el movimiento de los brazos debe realizar un barrido o arrastre hacia afuera.<sup>7</sup> También el movimiento de las piernas está dividido en tres fases que permiten que el pataleo sea efectivo.<sup>8 4</sup>

### 6.1.3. Nado de espalda

En este estilo de nado, el cuerpo se coloca en una posición más lateral que en estilo Crol, con la línea del agua justo debajo de las orejas y la barbilla sumergida en el agua. El pecho debe estar alineado con la superficie del agua y la cadera ligeramente debajo.

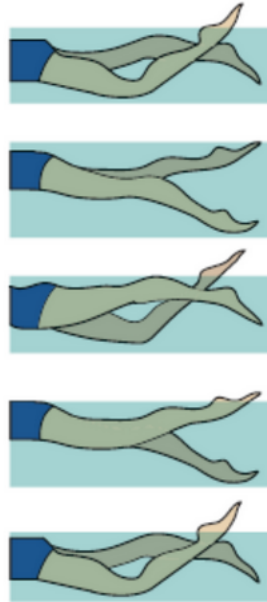


Figura 6: Acción y coordinación de piernas en el estilo Crol.



Figura 7: Movimiento de arrastre en estilo Braza de Pecho.

En este estilo es muy común las inclinaciones laterales del cuerpo y un barrido lateral de los brazos, sumergidos en el agua. Esto se puede apreciar mejor viendo la Figura 9 en donde se observa el movimiento del cuerpo en este tipo de nado. [5]

Así como los mencionados anteriormente existen otros estilos de natación que tienen su técnica individual y que son utilizados específicamente por sus características en distintas competiciones olímpicas y de alto rendimiento.

#### 6.1.4. ¿Qué es necesario medir en la natación?

Cuando un atleta o entrenador desea medir progreso del entrenamiento y competencia, es importante saber qué es lo que se desea medir. Cuando se está hablando de la disciplina de la natación hay muchos aspectos que se deben de tomar en cuenta, según Blythe Lucero en su libro llamado *Technique Swim Workouts* [6], la eficiencia es uno de los aspectos más importantes a medir para cualquier persona que desea perfeccionar su técnica en este deporte. ¿Pero cómo medir el progreso?

Algunas de las mediciones que se hacen comúnmente para determinar el progreso es la distancia total que el nadador puede recorrer, otra opción de medir progreso es establecer la distancia que el atleta es capaz de recorrer en un tiempo predeterminado, estos dos tipos de mediciones pueden archivarse y ser comparadas con el pasar del tiempo con las anteriores

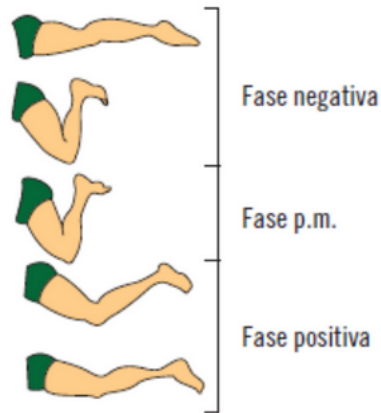


Figura 8: Movimiento de pateo en estilo Braza de Pecho.

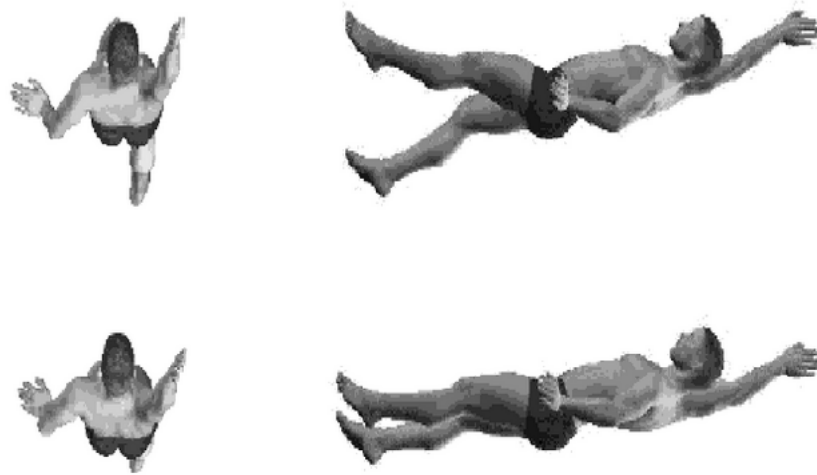


Figura 9: Movimiento de cuerpo en nado de espalda.

para determinar si el progreso ha sido positivo o no. Así como las dos opciones anteriores, hay muchas maneras de medir el progreso de los atletas y se asume entonces que si el progreso es positivo entonces la técnica ha mejorado en caso contrario se asume lo contrario pero por lo general esto se confronta con más entrenamiento y no científicamente.

Con lo mencionado anteriormente sobre los estilos de natación y cómo la técnica es de gran importancia para su correcta implementación y eficacia se puede decir también que es necesario obtener mediciones que permitan definir que tan bien se ejecuta la técnica dependiendo del estilo que se está utilizando.

## 6.2. La natación, un deporte olímpico

La natación se remonta a la edad de piedra, se aprendió por el ser humano debido a la necesidad de cruzar cuerpos de agua y de la pezca para obtener alimentos. Se conoce un poco de esta historia ya que se han encontrado glifos y pinturas en ruinas de Egipto y referencias en la mitología griega. La natación no se le conocía como un deporte y mucho menos un deporte organizado hasta el siglo XIX cuando en Gran Betaña, la Asociación de Natación inició con competencias de este deporte.

En los juegos olímpicos la natación fué incorporada en el año 1896 con cuatro eventos de competencia de nado Crol, posteriormente a partir del año 1904 se agregó otros tipos de nado como el nado de pecho y posteriormente de mariposa. En el año 1912 la natación femenina fue incorporado también como un deporte olímpico oficialmente. [7] Actualmente se cuentan con 17 eventos, tanto para categorías femenina y masculina, con las mismas características, con la única diferencia que para la libre de hombres se recorren 1500 metros y en el de mujeres 800 metros. En el Cuadro [1] se pueden observar las diferentes competencias que se presentan en la actualidad en los juegos olímpicos. [8]

Numero	Masculino	Femenino
1	50 metros estilo libre	50 metros estilo libre
2	100 metros estilo libre	100 metros estilo libre
3	200 metros estilo libre	200 metros estilo libre
4	400 metros estilo libre	400 metros estilo libre
5	1500 metros estilo libre	800 metros estilo libre
6	100 metros de espalda	100 metros de espalda
7	200 metros de espalda	200 metros de espalda
8	100 metros de pecho	100 metros de pecho
9	200 metros de pecho	200 metros de pecho
10	100 metros de mariposa	100 metros de mariposa
11	200 metros de mariposa	200 metros de mariposa
12	200 metros combinado	200 metros combinado
13	400 metros combinado	400 metros combinado
14	4x100 metros de relevos libre	4x100 metros de relevos libre
15	4 x 200 metros de relevos libre	4 x 200 metros de relevos libre
16	4 x 100 metros de relevos combinados	4 x 100 metros de relevos combinados
17	Maratón de 10 km	Maratón de 10 km

Cuadro 1: Competencias de natación en las olimpiadas

### 6.2.1. Récor ds de la natación en los Juegos Olímpicos

Una de las formas más sencillas para encontrar a los nadadores más veloces es observar los records mundiales que ocupan en las olimpiadas. En la tabla a continuación se recopiló algunos de los records actuales en algunas competencias de natación [2] que incluyen estilo libre, de espalda, pecho y mariposa. Si se observa con cuidado y se hace una relación sencilla de distancia recorrida y tiempos, es posible observar que la velocidad por lo general es menor

a los 3 m/s. Por razones lógicas, algunas competencias tienen velocidades más lentas. [9](#)

Nombre	Nacionalidad	Competencia	Tiempo
César Cielo Filho	Brasil	50 metros estilo libre	20.91
César Cielo Filho	Brasil	100 metros estilo libre	46.91
Missy Franklin	USA	200 metros de espalda	2:04.6
Lilly King	USA	100 metros de espalda	1:04.13
Katie Ledecky	USA	400 metros estilo libre	3:56.46
Katie Ledecky	USA	800 metros estilo libre	8:04.79
Liu Zige	China	200 metros mariposa	2:01.81
Kyle Masse	Canada	100 metros de pecho	57.13
Michael Phelps	USA	100 metros mariposa	49.82
Michael Phelps	USA	200 metros mariposa	1:51.51
Sarah Sjöstrom	Suecia	50 metros estilo libre	23.67
Sarah Sjöstrom	Suecia	100 metros estilo libre	51.71
Sarah Sjöstrom	Suecia	100 metros mariposa	55.48

Cuadro 2: Records de natación actuales

### 6.3. Visión de Computadora

La visión de computadora es un campo de la inteligencia artificial que permite interpretar y analizar el mundo visual de forma independiente y tomar decisiones a partir de la información que colecta. En esta se capturan datos de una imagen fija o video y se realiza una acción o se toma una decisión para implementarla en el mundo real. Por lo general la data obtenida tiene información para contextualizarla que permite al sistema tomar una mejor decisión. Aún así en un sistema de visión de máquina la computadora recibe un arreglo de números que debe interpretar y procesar para poder construir una instrucción o patrón a seguir.

Luego de entender qué es la visión de computadora, uno se podría preguntar, ¿Para qué se puede utilizar? No sería poco acertado asegurar que la visión de computadora pronto estará en todos lados, esta permite a las computadoras obtener información útil sobre el funcionamiento de las máquinas, como por ejemplo los carros inteligentes, las fábricas y los equipos semi-autónomos, así como también sobre el comportamiento de las personas hacia ciertas situaciones específicas. La visión de computadora no se detiene en el manejo y control de maquinaria, en la era digital que se está viviendo, en donde se comparten videos y fotografías masivamente por toda la red, las diferentes aplicaciones y servidores utilizan algoritmos de reconocimiento que permiten obtener información de todo este material compartido. [10](#)

### 6.4. PyQt

**PyQt** es un conjunto de uniones entre Python y la aplicación Qt creadas por la compañía River Bank Computing. **Qt** es un conjunto de librerías en C++ que permiten, de una manera sencilla, desarrollar interfaces gráficas de usuario, Bases de Datos SQL, OpenGL, XML,

animaciones 3D, etc. La versión más nueva de PyQt (PyQt5) implementa más de 1000 de estas clases como un set de módulos en Python. [\[11\]](#)

PyQt5 puede ser utilizado en Windows, Linux, UNIX, Android, macOS y plataformas iOS y existe algunas plataformas como QtDesigner que permiten de forma gráfica desarrollar la aplicación y posteriormente transcribirla a Python u otro lenguaje en el que se desee utilizar. PyQt5 cuenta con algunos programas de utilidades que pueden ser importantes:

- **pyuic5**: que convierte Widgets de Qt, creados utilizando herramientas de Qt como Qt Designer, a código de Python.
- **pyrcc5**: se encarga de incrustar algunos recursos como imágenes, íconos, archivos, etc.
- **pylupdate5**: extrae información de Python que necesita ser traducida y crea o actualiza archivos como el .ts.

## 6.5. OpenCV

OpenCV es una librería de visión de computadora de fuente abierta. Esta librería fue diseñada para ser portable por lo que puede ser compilado en varios lenguajes de programación y está disponible para varios sistemas operativos como Linux, macOS y Windows. Esta librería fue diseñada para aplicaciones en tiempo real y optimizada para eficiencia computacional tomando ventajas de los procesadores de las computadoras y si las características de la computadora lo permiten este puede aprovecharlas para utilizar procesadores multinúcleo. [\[12\]](#)

La estructura a gran escala de OpenCV está dividida en cuatro componentes básicos, mostrados en la Figura [\[10\]](#) en donde:

1. **CV**: es el componente que contiene las librerías de procesamiento de señales y algoritmos de visión de computadora.
2. **MLL**: este componente contiene las librerías de aprendizaje de computadora (Machine Learning).
3. **HighGUI**: este componente contiene las rutinas de entrada/salida y funciones para almacenar y cargar imágenes y videos.
4. **CXCORE**: este componente contiene las estructuras básicas de data, contenido y funciones.

## 6.6. Procesamiento de imágenes

El procesamiento de imágenes es una disciplina que viene ligada con el procesamiento de señales. Por lo general este se utiliza para filtrar imágenes, mejorar su apariencia visual o

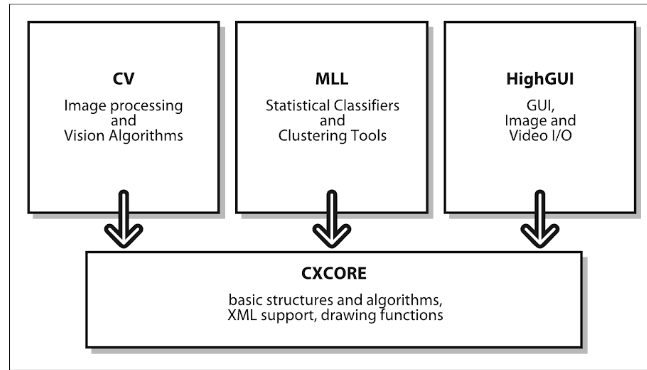


Figura 10: Estructura básica de OpenCV.

12

simplemente para prepararlo para algún otro proceso posterior. A diferencia del estudio de la visión del ser humano esta disciplina es útil en el momento en que se tiene la capacidad para manipular y almacenar gran cantidad de información. 13

El procesamiento digital de imágenes se puede dividir en cuatro pasos que son, capturar la imagen, realizarle mejoras visuales, la segmentación y la extracción de información.

1. **Captura de imágenes:** durante este proceso se obtiene la imagen por cualquier medio, dependiendo de la aplicación este proceso puede ser tomar una imagen con una cámara sencilla pero puede llegar a ser tan complejo como utilizar cámaras con sensores especializados.
2. **Mejoras de la imagen:** por lo general, al capturar una imagen esta no está en condiciones óptimas para ser procesada, por esta razón se procede a realizar recortes a un tamaño deseado y en caso sea necesario se filtra la imagen para reducir ruido, eliminar algún factor no deseado o colorear la imagen.
3. **Segmentación de la imagen:** por la misma razón de que la imagen capturada puede que no esté en las condiciones deseadas, se procede a realizar la etapa de segmentación que permite simplificar la representación de la imagen hasta aislar regiones u objetos de interés.
4. **Extracción de información:** esta última etapa es la que se encarga de encontrar y localizar patrones geométricos o visuales que se desean detectar y posteriormente evaluar.

## 6.7. Filtrado de imágenes

El filtrado de imágenes, como se menciona anteriormente, se utiliza generalmente para realizar mejoras a las imágenes antes de procesarlas. En el filtrado de imágenes el valor de un píxel en la salida de una imagen está dado por un algoritmo que altera los valores en sus vecindades. El filtrado lineal de imágenes se puede lograr por medio de la convolución, en

donde se toma el peso de la suma de los valores de sus vecinos. A esta matriz de pesos se le llama **Kernel**.<sup>[14]</sup>

Existen infinidad de filtros que se pueden utilizar y estas dependen del uso que se les va a dar. Una de las grandes ventajas que se tiene con OpenCV, es que cuenta con funciones integradas que permiten implementar filtros a imágenes con comandos sencillos, a continuación se describen algunos filtros que podrían ser de utilidad.<sup>[15]</sup>

### **Blur()**

Se puede definir un Box Blur, como un difuminado básico, en esta operación todos los pesos son colocados en 1. La ventaja de esta operación es su rapidez y su poco costo en programación. Un ejemplo de este filtro se muestra en la Figura <sup>[11]</sup>



Figura 11: Ejemplo del Box Blur

<sup>[16]</sup>

Esta función de Opne CV difumina una imagen utilizando el Box Filter. Tiene como argumentos la imagen de entrada, la imagen de salida, el tamaño del kernel, el centro del kernel y la extrapolación de los píxeles. <sup>[15]</sup>

### **GaussianBlur()**

Un filtro Gaussiano es básicamente un filtro pasa-bajas que implementado se calcula independientemente para cada píxel de la imagen y se puede utilizar la ecuación a continuación para calcular los valores de salida <sup>[17]</sup>:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2 + y^2}{2\sigma^2}} \quad (1)$$

En donde  $\sigma$  es la desviación estándar y  $x$  y  $y$  son las distancias originales a los ejes. Un ejemplo claro de este tipo de filtrado se puede observar en la Figura <sup>[12]</sup>

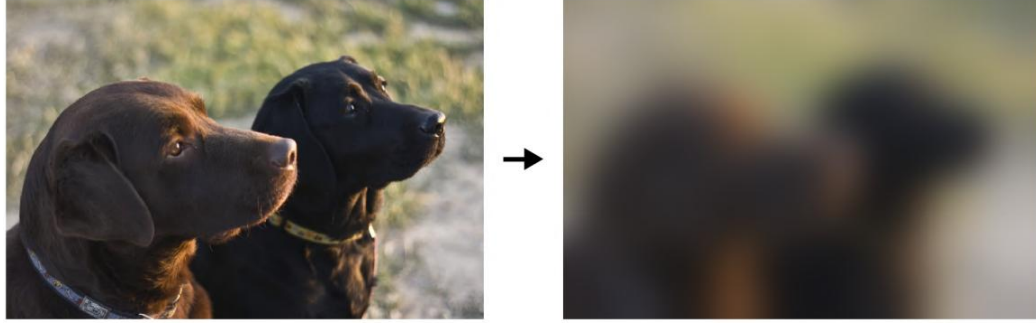


Figura 12: Ejemplo de filtro Gaussiano

[18]

Esta función de OpenCV difumina una imagen utilizando el filtro Gaussiano. Tiene como argumentos la imagen de origen, la imagen de salida, el tamaño del kernel, la desviación del kernel en "x" y "y" y el método de extrapolación del píxel.

## 6.8. Imágenes digitales

Por lo general, las imágenes se representan con modelos matemáticos, cuando estas funciones o variables tienen un significado físico se les conoce como señales. Una imagen monocromática puede ser descrita por medio de número escalares pero las imágenes a color deben de ser representadas por medio de vectores con información de los colores.

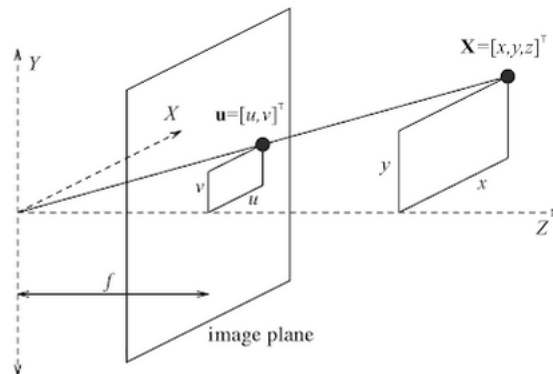


Figura 13: Proyección de una imagen.

Las imágenes en el mundo real son tridimensionales pero nuestros ojos o los sensores como las cámaras los traducen a una proyección de intensidad de luz en dos dimensiones modelada por una cámara de agujero como se ilustra en la siguiente Figura [13], en donde se evidencia la reflexión respecto al plano XY en donde las variables  $x$ ,  $y$  y  $z$  representan coordenadas en el plano 3D y  $f$  es la distancia desde el agujero hasta el plano de la imagen (longitud focal). El punto proyectado  $u$  tiene coordenadas  $(u, v)$  que pueden ser determinadas con las siguientes expresiones derivadas de triángulos semejantes. [19]

$$u = \frac{xf}{z}v = \frac{yf}{z} \quad (2)$$

$$v = \frac{yf}{z} \quad (3)$$

Una imagen digital es una imagen  $f(x, y)$  discretizada en coordenadas espaciales y de brillo formando series de arreglos, uno por cada banda de color. [20]

## 6.9. Thresholding y extracción de información

El thresholding es una técnica utilizada para hacer binaria las imágenes basado en los límites de color o de brillo de la imagen. Esta suele ser una técnica muy utilizada ya que no requiere de mucha capacidad computacional y su aplicación puede ser tan simple como delimitar el rango de colores de una imagen.

La implementación más común del threshold es al evaluar todos los píxeles de la imagen y sustituirlos por un píxel completamente negro si estos superan un nivel de luminosidad preestablecido. A pesar de que este método es muy sencillo de implementar y muy bueno para su cometido existen otros métodos que permiten tomar en cuenta otros para extraer información y segmentar la imagen. Algunos de estos pueden ser basados en las formas e histogramas, basados en las esquinas y combinación del gradiente entre el final de la figura y el fondo y otros que usan algoritmos para determinar basados en el entorno y la imagen original un valor de threshold adecuado.

## 6.10. Visual Servoing

El control basado en la visión, también conocido como visual servoing es un tipo de control basado en la visión por lo que se puede decir que es utilizar información visual para controlar la pose del efector final de un robot relativa al objetivo. Hay dos tipos posibles de realizar visual servoing, basándose en la posición (PBVS por sus siglas en inglés) y el basado en imágenes (IBVS por sus siglas en inglés). [21]

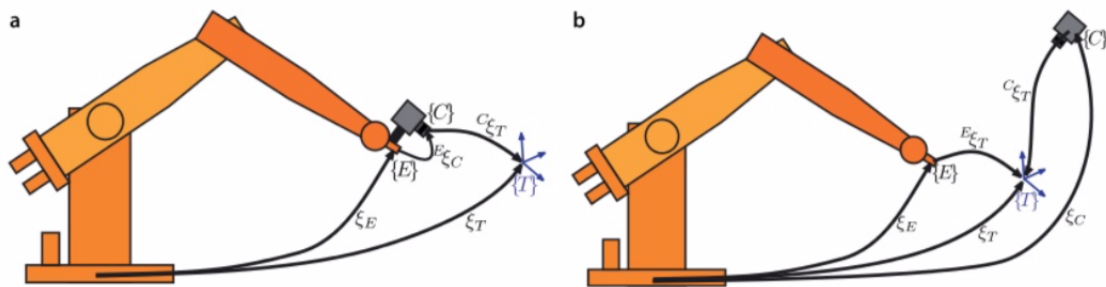


Figura 14: Diferencia entre el PBVS y el IBVS.

[21]

### 6.10.1. Position Based Visual Servoing (PBVS)

En el caso del control visual basado en la posición la pose del objetivo respecto al sensor (la cámara) es estimada, esto quiere decir que para poder realizar los cálculos es necesario tener información sobre la geometría del objetivo, parámetros intrínsecos de la cámara y algunas características del plano de la imagen. El control basado en la posición de un sistema simple se puede describir con el diagrama de bloques mostrado en la figura a continuación, en donde se puede observar que el controlador del movimiento depende directamente del controlador de posición.[15](#)[14](#)

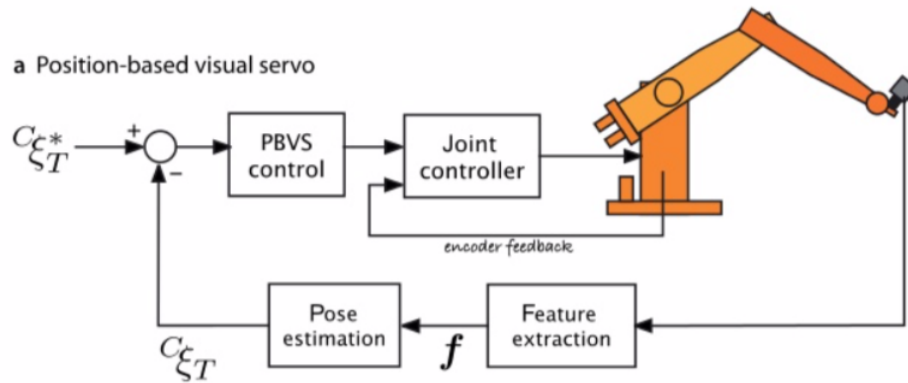


Figura 15: Sistema controlado por un PBVS.

[21](#)

### 6.10.2. Image Based Visual Servoing (IBVS)

El control visual basado en imágenes, a diferencia del basado en posición, no estima la pose del objetivo ya que esta está implícita en los valores de las características de la imagen. En este caso el problema de control puede ser planteado en coordenadas de la imagen. En pocas palabras se puede decir que en este caso se cambió el problema de estimar una pose a controlar puntos de una imagen. Un sistema controlado por este tipo de control se puede describir sencillamente con un diagrama de bloques como el mostrado en la figura a continuación[16](#) en donde se puede evidenciar a comparación del diagrama de bloques del PBVS [15](#) que no realiza ninguna estimación de la pose en la retroalimentación.

## 6.11. Raspberry Pi 3

La Raspberry Pi es una computadora de bajo costo desarrollada en el Reino Unido con el objetivo de enseñar computación básica en el año 2009. Esta es una herramienta, impresa en una sola placa, que permite a miles de personas interactuar con el funcionamiento de una computadora.

La Raspberry Pi 3 B es un modelo reciente de esta pequeña computadora, que, a diferencia de sus predecesores, cuenta con un procesador ARM (1.2 GHz) con arquitectura SoC.

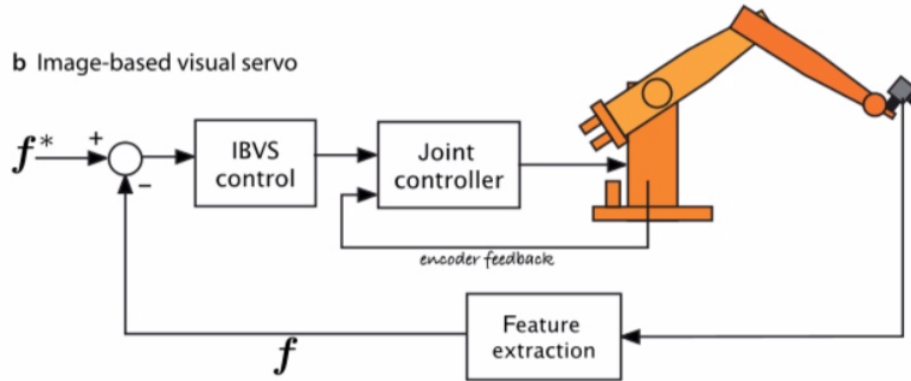


Figura 16: Sistema controlado por IBVS.

[21]

[1] También cuenta con la opción de colocar una memoria Micro SD con el sistema operativo y en donde se pueden almacenar otros archivos ya que la memoria interna de la Raspberry Pi 3 es de tan solo 1GB. [22] Algunas especificaciones importantes de la Raspberry Pi 3 Modelo B se pueden observar en la tabla a continuación [3], [23]

<b>Arquitectura</b>	ARMv8
<b>SoC</b>	BCM2837
<b>CPU</b>	1.2GHz 64-bit Quad-core
<b>GPU</b>	Dual-core VideoCore 4 3D
<b>RAM</b>	1GB LPDDR2
<b>USB</b>	4 x USB 2.0
<b>Salida de Video</b>	Full HDMI port
<b>Almacenamiento</b>	Micro SD
<b>Network</b>	Wireless LAN 10/100 Ethernet (RJ45)
<b>Bluetooth 4.1</b>	Fuente de Poder & 5 Volteos (Micro USB)

Cuadro 3: Características técnicas de la Raspberry Pi 3

<sup>1</sup>Sistem on Chip

---

## Diseño mecánico del sistema

---

El prototipo del "*Sistema Autónomo de Seguimiento de Atletas en la Natación por Medio de una Cámara*" consiste en el diseño de un sistema guiado por medio de algún tipo de riel y que permite montar en él una cámara de video para desplazarla horizontalmente a la orilla de una piscina y poder de esta forma dar un seguimiento apropiado a los nadadores. En el capítulo presentado a continuación se describe y se discuten las distintas fases y prototipos construidos y utilizados para validar el diseño de dicho sistema.

### 7.1. Requisitos del sistema

Como primer paso para el diseño de los prototipos presentados a continuación, se definió un listado de tres requisitos generales que permitan validar el sistema de una manera objetiva. El listado de dichos requisitos se muestra a continuación:

- El sistema debe ser capaz de desplazarse horizontalmente.
- El sistema debe permitir que se monte, como mínimo, una cámara para ser desplazada.
- El diseño del sistema debe permitir que este sea montado cercano a una piscina.

A parte de cumplir estos requisitos se busca que el sistema sea portable y pueda ser armado por cualquier persona que no tenga conocimientos técnicos.

## 7.2. Primer prototipo

Para el primer prototipo del sistema mecánico se diseñó y construyó un dispositivo que consistió en dos rieles de madera de pino por encima del nivel del suelo, en dos bases fabricadas en MDF, en el cual se montó un carrito que por medio de un mecanismo con cojinetes se desplazaría a lo largo de dichos rieles. Posteriormente se procedió a validar el sistema para verificar que los requisitos establecidos se cumplieran.

### 7.2.1. Diseño del prototipo

Para el diseño del primer prototipo se procedió a investigar sobre sistemas existentes que presenten características similares a las deseadas. La principal inspiración de este prototipo es un slider utilizado para montar en el cámaras de video profesional [17](#). De este se tomó la idea de utilizar dos rieles como guía para desplazar un carrito, que por medio de un juego de rodamientos, se desliza horizontalmente a través de él. Para concretar el diseño se procedió a realizar un modelo en un software de modelado en 3D [1](#) como se muestra en la Figura [18](#), para posteriormente definir los materiales y las piezas que serían necesarias para su construcción.

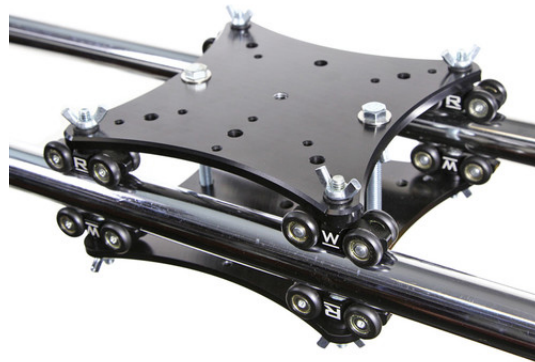


Figura 17: Prototipo 1: Slider utilizado como inspiración.

[24](#)

### 7.2.2. Construcción del prototipo

Posterior al diseño del sistema de este primer prototipo se procedió a conseguir los materiales, las piezas mecánicas y la manufactura de las piezas necesarias [4](#). Para los rieles se utilizó un tubo de madera de pino con dos metros de largo y una pulgada de diámetro, este se partió justo por la mitad para conseguir dos tubos de 1 metro cada uno. Al tener

---

<sup>1</sup>Autodesk Inventor 2018

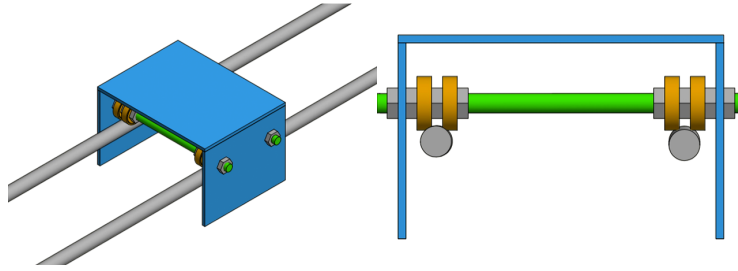


Figura 18: Prototipo 1: Modelo 3D

los dos tubos, que serían los rieles, se procedió a diseñar una base que permitiera montarlos sobre el nivel del suelo. Las bases se diseñaron y fueron maquinadas posteriormente en una cortadora láser en MDF de 5 mm de espesor y luego ensambladas con pegamento blanco para madera y tornillos como las mostradas en la Figura 19.

Materiales	Cantidad	Dimensión	Comentarios
Perfil Circular Madera de Pino	1	Diámetro: 1 pulgada Largo: 2 metros	Se partió a la mitad para obtener 2 de 1 metro cada uno.
Plancha MDF	1	30 x 40 cm	
Tornillos	8	M 3	
Tuercas	8	M 3	

Cuadro 4: Materiales para prototipo 1, sistema de rieles.

Es importante mencionar que inicialmente las patas de esta base no contaban con un soporte en la parte inferior lo cual causaba que este no fuera estable y se balanceaba de lado a lado, por lo que se procedió a iterar este diseño con un pequeño soporte triangular en cada una de las patas, como los que se observan en la Figura 19. Posteriormente para el ensamblaje de los dos rieles y las patas de soporte se utilizó tornillos de 1/8 y con estos se logró concluir con la construcción del sistema de rieles como se muestra en la Figura 20. Una vez concluido el sistema de rieles se procedió a construir el carrito que sería montado para deslizarse sobre ellos. Para la construcción del carrito se diseñó y se cortó, nuevamente en MDF de 5 mm de espesor, las piezas que conforman las paredes superior y laterales. En las paredes laterales se realizó dos agujeros en los cuales se montó los sistemas de rodamientos como los mostrados en la Figura 21. Los materiales utilizados para la construcción de este se detallan en la Tabla 5.

### 7.2.3. Pruebas de validación del prototipo

Una vez construida esta primera fase del prototipo se procedió a evaluar el primer requisito planteado, el cual establece que el sistema debe ser capaz de desplazarse horizontalmente. Para evaluar el primer requisito presentado se montó el carrito en el sistema de rieles



Figura 19: Prototipo 1: Base del sistema



Figura 20: Prototipo 1: Rieles de madera

<b>Materiales</b>	<b>Cantidad</b>	<b>Dimensión</b>	<b>Comentarios</b>
Tubo roscado M 8	2	Diámetro: 8 mm	
Roldana	16	8 mm	
Rodamiento	8	Diámetro exterior = 22 mm Diámetro interior = 8 mm	
Tuercas	8	M 3	

Cuadro 5: Materiales para prototipo 1, carrito.

y se deslizó con las manos para verificar que efectivamente este se moviera horizontalmente. Con esta sencilla prueba se pudo comprobar que se cumplió con una parte del primer requerimiento, posteriormente se procedió a dar un pequeño empujón al carrito para intentar desplazar de un lado del sistema hacia el otro, al intentar esto se fue comprobando que este no se podía mover libremente como se deseaba. Se consideró que la fricción de la madera es un factor que afecta en el deslizamiento a pesar que los perfiles se lijaron para reducir este factor.

Posteriormente al realizar observaciones más detenidas se pudo observar que existe

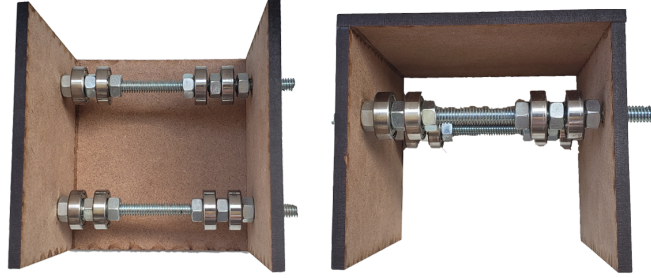


Figura 21: Prototipo 1: Carrito 1.

un pequeño pandeo entre los dos tubos, lo cual hace que el carrito se detenga ya que la separación entre los rieles era menor en el centro del sistema que en los extremos, por lo que en esta sección del trayecto se movía con más libertad que a los costados. Esto se consideró como un problema que se podría resolver al cambiar el material a utilizar para los rieles, sin embargo es un problema que se podría dar nuevamente al utilizar perfiles extremadamente largos como los que se necesitarían en caso se construyera del largo de una piscina olímpica o semi-olímpica. Por esta razón se procedió a descartar este prototipo y no se continuó intentando cumplir con los otros requisitos ya que se procedió a iterar con un nuevo diseño.

### 7.3. Segundo prototipo

Debido a los problemas encontrados en las primeras etapas de validación del primer prototipo, se procedió a evaluar posibles cambios para encontrar una solución. Por esta razón se re diseñó el modelo mecánico principal utilizado anteriormente. El nuevo modelo planteado en este prototipo continua con la misma idea de utilizar un riel como guía, pero en este caso se utiliza solamente uno. Para el carrito utilizado se cambió el sistema de deslizamiento y se colocó cuatro llantas que complementarían en el equilibrio del mismo.

#### 7.3.1. Diseño del prototipo

Siempre siguiendo el camino de intentar cumplir con los requisitos principales planteados en un inicio, se procedió a realizar un nuevo diseño del sistema, manteniendo el concepto principal pero realizando algunos cambios como se menciona anteriormente. En este nuevo diseño se basó principalmente en un sistema lineal de rodamientos como el mostrado en la Figura 23, en donde se puede observar que el cable pasa por el medio de tres cojinetes que le permiten deslizarse libremente pero a la vez permiten utilizar el cable como una guía o riel sin generar mucha fricción. Este sistema lineal de rodamientos se colocó en la parte central de un carrito, similar en forma al diseñado para el primer prototipo. Adicional se le agregó ocho llantas (cuatro interiores y cuatro exteriores como se muestra en la Figura 23) que brindarían equilibrio al sistema, ya que al solamente tener un riel como guía este podría

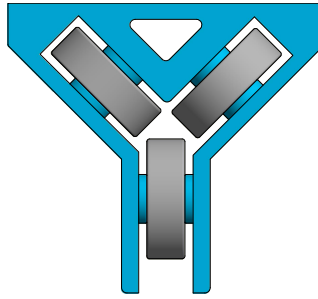


Figura 22: Prototipo 2: Sistema de cojinetes

fácilmente balancearse hacia cualquier lado y topar con la superficie. A diferencia del carrito utilizado en el prototipo 1, a este se le colocó los agujeros para el eje de cada una de las llantas y agujeros para montar el sistema de cojinetes.

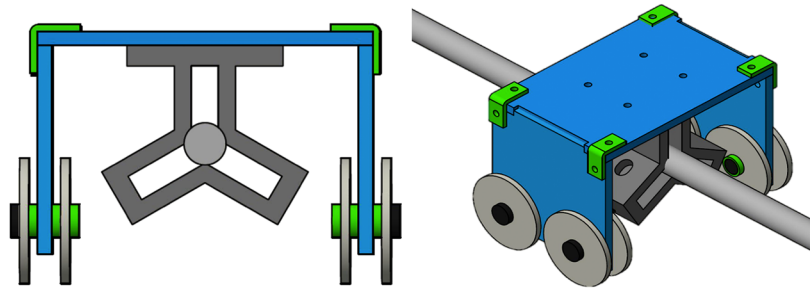


Figura 23: Prototipo 2: Modelo 3D

### 7.3.2. Construcción del prototipo

Para la construcción de este prototipo se inició con la manufactura del sistema lineal de rodamientos, este se fabricó en MDF de 3mm. Para esto se imprimió 12 copias del modelo y se pegó una con otra para darle un mayor espesor, como se muestra en la Figura 24. Posteriormente se procedió a realizar los agujeros necesarios para poder atornillar los rodamientos en su lugar, esto se realizó con un barreno y una broca para madera de 1/8" inicialmente y posteriormente con una de 5/16 ". Al realizar esto el modelo falló y se destruyó como se muestra en la Figura 25. Después del fallo del primer intento se tomó la decisión de imprimir la pieza en PLA utilizando una impresora 3D, se utilizó el mismo modelo que inicialmente se había planteado.

Al tener la pieza impresa, como se muestra en la Figura 26, se procedió a colocarle



Figura 24: Prototipo 2: Mecanismo de rodamientos MDF



Figura 25: Prototipo 2: Fallo mecanismo de rodamientos

los rodamientos en los espacios correspondientes. Simultáneamente a esto se cortó con una cortadora láser en MDF de 3 mm de espesor las paredes y llantas del carrito. Las paredes se pegaron con pegamento blanco para madera. Para las llantas se imprimió también ejes en PLA y se cortó unas pequeñas roldanas en la cortadora láser en MDF de 3 mm de espesor.

### 7.3.3. Pruebas de validación del prototipo

En el momento en que este prototipo se estaba construyendo se encontró varios problemas que no permitirían que este cumpliera con los requerimientos planteados inicialmente, por lo que no fue posible realizar pruebas de validación para comprobar su funcionamiento. El primer problema que se encontró, fue el mencionado anteriormente en el cual al construirlo con MDF este falló, por lo que esta dejó de ser una opción para este tipo de mecanismos. Posteriormente al haber impreso la pieza se encontró un nuevo problema ya



Figura 26: Prototipo 2: Mecanismo de rodamientos PLA

que los rodamientos no giraban libremente porque tenían contacto directo con las paredes, esto nuevamente afectaría en el libre desplazamiento del sistema a la hora de montarlo sobre su riel guía. Sin embargo, con la construcción de dos prototipos completamente diferentes, ya se tenía una idea más clara de lo que se buscaba y cómo esto se podría conseguir. Por esta razón se decidió proceder a re diseñar algunos aspectos de este prototipo, como por ejemplo el sistema de rodamientos y la forma en que se desplazaría el carrito.

## 7.4. Tercer prototipo

Para el diseño de este tercer prototipo se tomó en consideración todos los problemas que se tuvo anteriormente y se tomó en cuenta también otros aspectos que antes no se habían considerado, como la necesidad de montar una cámara y los motores. Para este último prototipo se siguió trabajando bajo la misma idea con la que se venía trabajando y adicionalmente se le incorporó los motores que lo harían desplazarse y el sujetador para la cámara, dos cosas que en los diseños anteriores no se habían considerado un su totalidad.

### 7.4.1. Diseño general del prototipo

Para el diseño de este último prototipo se tomó la misma idea de un carrito con cuatro llantas que serían colocadas con un eje de madera en las paredes laterales, como se muestra en la Figura [27](#). Adicional a esto se agregó el sujetador para colocar los motores, que se discuten posteriormente en este documento. En la parte superior del carrito se colocó un sujetador para un tubo de ocho milímetros de diámetro, el cual será utilizado para sujetar las cámaras. Por último, se utilizó un sistema de solo un rodamiento ya que se consideró y se comprobó posteriormente que no era necesario más.

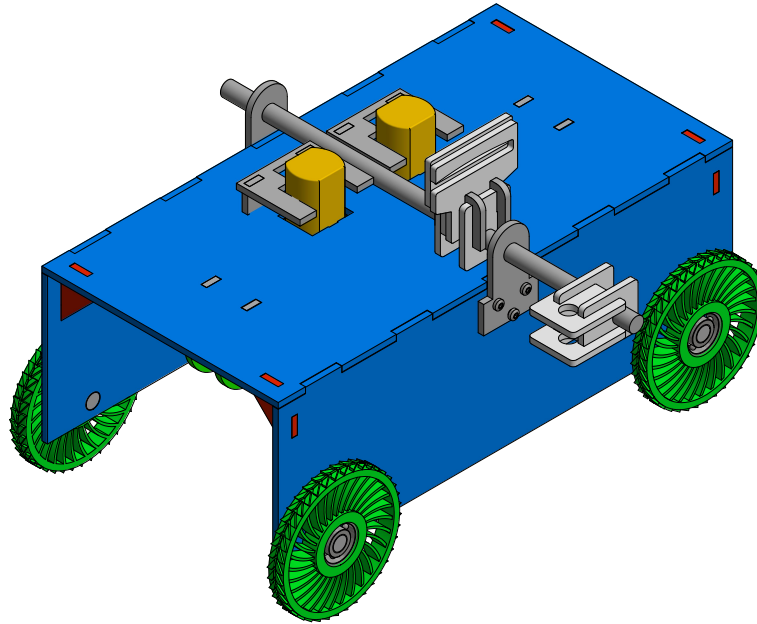


Figura 27: Prototipo 3: Modelo 3D

#### 7.4.2. Diseño de llantas

Para el diseño de las llantas (se puede observar en la Figura [29](#)), que se utilizó en este último prototipo, se tomó en consideración varios aspectos como el peso y tamaño de la llanta, su textura y el material del que esta sería fabricada.

El peso y tamaño de la llanta son dos factores importantes, ya que no se desea que estas agreguen mucho peso al diseño, por lo que cada una de las llantas debe ser liviana y no muy grande, en relación a las dimensiones del carrito. Otro aspecto importante a tomar en cuenta, fue la tracción. Las llantas en este prototipo son para brindar equilibrio y no tracción, por esta razón el motor no está conectado a ellas de ninguna forma, sin embargo se agregó un simple relieve a las llantas para que sin importar la superficie en que estén, estas rueden y no se deslicen por la falta de fricción. Por último, también se buscó que el material de las llantas no fuera muy duro, para que a la hora de deslizarse por superficies difíciles estas no generen vibraciones no deseadas en el carrito, que afecten en la grabación del video.

Se diseño una llanta y se realizó un modelo en MDF (utilizando la cortadora láser), como el que se muestra en la Figura [28](#) para comprobar que cumpliera con las especificaciones de tamaño necesarias para el prototipo. Luego se procedió a fabricar de varios materiales para comprobar las diferencias entre los resultados. Estos diferentes modelos se comparan en la Figura [30](#) en donde el modelo 1 y 2 fueron impresos en una impresora 3D con tecnología FDM [2](#) y el modelo 3 y 4 por una impresora 3D con tecnología SLA [3](#). Al comparar los

<sup>2</sup>Modelo por deposición fundida por sus siglas en inglés, son el modelo más común de impresoras 3D.

<sup>3</sup>Esterografía, conocidas convencionalmente como impresoras de resina líquida.



Figura 28: Prototipo 3: Llanta MDF

cuatro modelos se pudo hacer evidente varias cosas:

1. **PLA:** La llanta fabricada con este material mostró un buen acabado pero es muy dura, por lo que a pesar de tener la textura en su superficie suele resbalarse y podría generar vibraciones indeseadas.
2. **TPU:** La llanta fabricada con TPU, aunque no mostró un acabado tan bueno como el PLA, tiene propiedades un poco más flexibles y menos duras, por lo que se considera un candidato viable.
3. **Resina dura:** Este material mostró el mejor acabado, no solo en detalles sino que es la menos pesada de las cuatro pruebas realizadas. El problema con este material nuevamente es que es duro, incluso se siente más duro que el PLA.
4. **Resina flexible:** Este material es muy complicado, como se observa en la Figura es muy maleable y se rompe con facilidad, por lo que no es una opción.

### 7.4.3. Desplazamiento horizontal

El desplazamiento horizontal del sistema ha sido el requisito que se ha complicado por cumplir en los prototipos expuestos anteriormente. Por esta razón no se dejó a un lado esta vez y se diseñó desde un inicio pensando en la manera en que se realizaría el movimiento. Como se puede observar en la Figura [31](#), en la vista inferior del sistema, se puede observar que a los motores se les colocó un pequeño piñón con picos (como el mostrado en la Figura [32](#)) en su eje para que este genere fricción con el cable en el momento en que tengan contacto, junto con unos topes para evitar que el cable se aleje de ellos. En los costados se encuentra un mecanismo sencillo con un cojinete que puede girar libremente, sobre el cual se coloca el cable para generar un desnivel respecto a los piñones colocados en el eje de cada uno de los motores, este mecanismo se puede observar en la Figura ??.

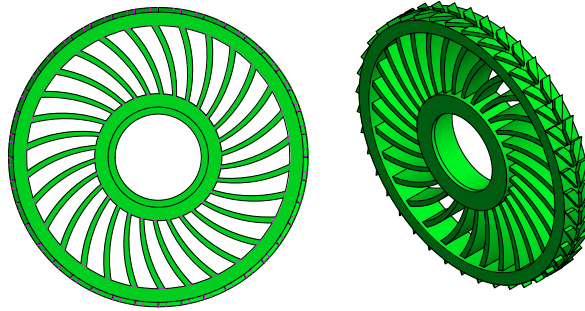


Figura 29: Prototipo 3: Lantas

#### 7.4.4. Diseño de sujeción de cámara

Uno de los requisitos del prototipo es que se pueda montar en el una cámara de video, para esto se diseñó una base como la mostrada en la Figura ?? en la cual se puede observar que es un sistema pensado para que una parte esté montada en el sistema y la otra, en la que se coloca la cámara, puede ser montada y desmontada en cualquier momento. Este mecanismo debe ir montado en el tubo que se coloca en el carrito, como el que se muestra en la Figura 31. Este pequeño sistema fue fabricado en MDF de 3 mm y ensamblado. Las medidas utilizadas para este se basan en la cámara con la que se cuenta para realizar las pruebas<sup>4</sup>, por lo que en el momento en que se desee cambiar la cámara es necesario cambiar las dimensiones del mismo.

#### 7.4.5. Bases para sujeción de cable

Uno de los principios más importantes para que este sistema funcione de la forma en que se plantea, requiere de un cable tensado como guía para el carrito. Para poder tensar un cable de manera sencilla se construyó dos cajones (20 x 15 cm) con MDF de 3 mm de espesor, estos se rellenaron con mezcla de concreto para brindar un peso suficiente y que estas no se desplazaran al oponerse a la fuerza del cable tensado. En cada uno de los cajones se colocó una argolla la cual sería utilizada para la sujeción del cable, la argolla se fundió en conjunto con el cajón para que esta quedara insertada en el bloque de material. Posteriormente se colocó en uno de los extremos del cable un tensor unido a una de las argollas en los cajones y el otro extremo se dejó fijo a la argolla del otro bloque.

#### 7.4.6. Construcción del prototipo

Se procedió a construir el sistema mencionado anteriormente. En su mayoría las piezas utilizadas para la construcción del carrito se fabricaron en MDF de 3 mm de espesor como

---

<sup>4</sup>WebCam Logitech C270 HD 720p USB

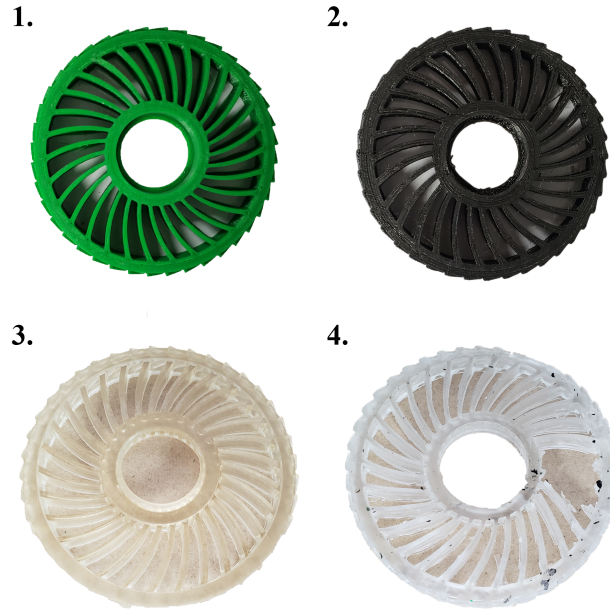


Figura 30: Prototipo 3: Comparación de llantas

se puede observar en la Figura 35 a excepción de las llantas, que se fabricaron con TPU como se discute anteriormente.

Para la construcción de los bloques utilizados para sujetar el cable tensado se utilizó MDF de 3 mm nuevamente para el molde y estos se rellenaron de mezcla de cemento para fundición, se colocó plástico en dentro de los cajones de madera ya que la mezcla de cemento podría humedecer y deformarlos. El resultado final de estos bloques de cemento se puede observar en la Figura 36 Se colocó los dos bloques con un metro de separación, se tensó el cable por medio del tensor y se colocó el carrito, asegurando de que el cable pasara tanto por debajo de los dos ejes de los motores como por encima de los dos mecanismos de cojinetes. Una vez realizado esto se procedió a desarrollar las pruebas de validación.

#### 7.4.7. Pruebas de validación del prototipo

Para comprobar el correcto funcionamiento del sistema y si este cumple con los requisitos planteados anteriormente, se procedió a instalar los motores en el sistema y se conectaron los mismos a corriente, intercambiando polaridades para cambiar la dirección de desplazamiento del carrito. Al realizar esto se pudo comprobar que definitivamente el carrito se desplaza horizontalmente en ambas direcciones. Posteriormente se procedió a realizar más pruebas, que se mencionan en el capítulo "Selección de Motor" para comprobar el rendimiento del mismo. Habiendo comprobado esto, se puede decir que se cumple el primero de los tres requisitos planteados.

Una vez se comprobó que el carrito se desplaza en ambas direcciones se procedió a instalar el mecanismo para sostener la cámara. Una vez colocada la cámara se realizó nuevamente pruebas de desplazamiento horizontal y se pudo comprobar que efectivamente la cámara no

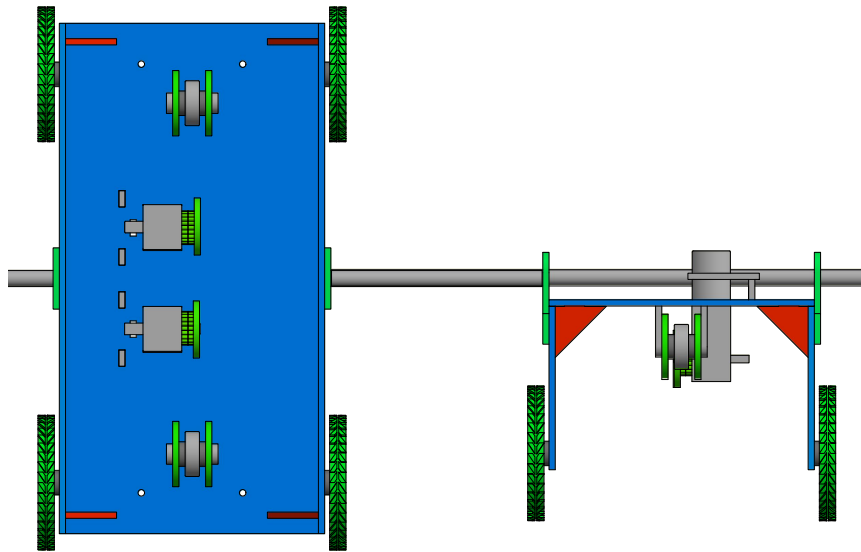


Figura 31: Prototipo 3: Vistas de prototipo final

se balancea ni se cae del sistema, por lo que con esto se puede decir que se cumple otro de los requisitos planteados. Más adelante en este documento se discute el movimiento del prototipo por medio del sistema de seguimiento y la interfaz gráfica de usuario. Por último por la forma en que se diseñó el sistema este puede ser colocado cercano a una piscina sin complicación alguna, se colocó a la orilla de una mesa, como se muestra en la Figura [37](#), con un tubo sujetado hacia abajo en donde se podría colocar otra cámara y este funcionó sin ninguna complicación. Por lo que se puede concluir según este diseño y a las pruebas realizadas que se cumple con los tres requisitos planteados en un inicio.

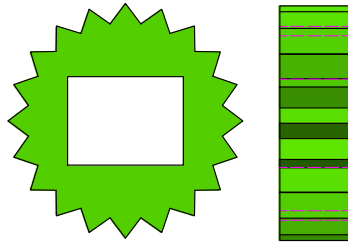


Figura 32: Prototipo 3: Piñón eje motor

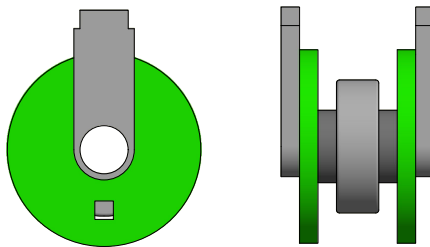


Figura 33: Prototipo 3: Mecanismo cojinete

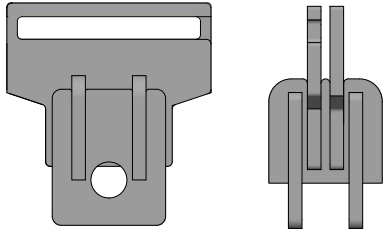


Figura 34: Prototipo 3: Sujeción de cámara

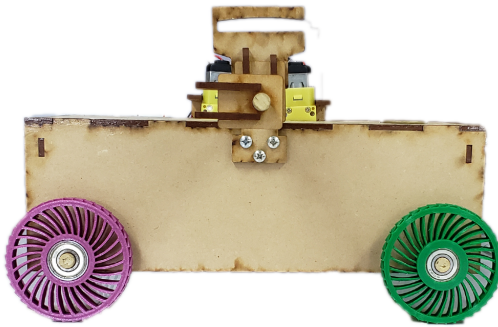


Figura 35: Prototipo 3: Carrito construido



Figura 36: Prototipo 3: Caja de cemento

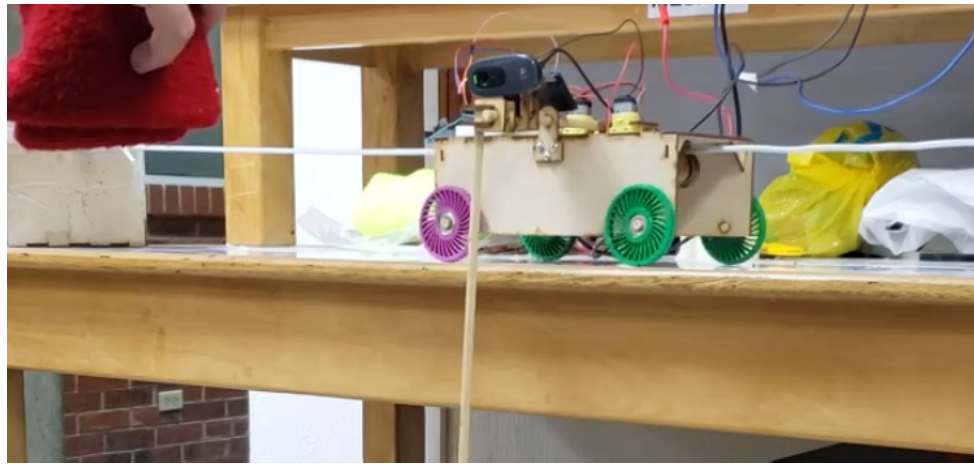


Figura 37: Prototipo 3: Funcionamiento

---

## Sistema de seguimiento por medio de una cámara de video

---

Como complemento al sistema mecánico es necesario generar una señal de control que maneja la dirección en que debe girar el motor que causa el movimiento de todo el sistema. Para esto se diseñó sistema que por medio de software toma imágenes de una cámara de video, detectar un objeto con un color específico y proporciona la dirección en que este se mueve para posteriormente transmitirla a dicho motor.

### 8.1. Requisitos del sistema

Como primer paso se estableció los tres requisitos principales bajo los cuales se trabajó durante el proceso y que permiten posteriormente validar si el sistema cumple con su función.

- El programa debe permitir conectar con una cámara de video y obtener imágenes por medio de esta.
- El programa debe obtener la posición de un objeto específico y determinar si este se desplaza y hacia qué dirección lo hace, para enviar al motor una señal de control válida.
- El programa debe poder transmitir una señal de video en, para obtener una retroalimentación de video en la aplicación.

### 8.2. Diseño de programa de seguimiento

Para el programa, que debe de cumplir con los requisitos mencionados anteriormente, se utilizó el lenguaje de programación Python 3 en conjunto con las librerías de OpenCv

para realizar el seguimiento de un objeto de color determinado, que puede ser seleccionado colocando como parámetros un límite inferior y superior en el espacio HSV y como sensor para la entrada de datos y para detectar el objeto del color deseado, se conectó una cámara de video.

Para obtener el video que sería evaluado para encontrar el objeto se capturó las imágenes utilizando la clase `VideoCapture()` con su función `read()`, la cual obtiene y decodifica el siguiente frame<sup>1</sup> del video de la cámara seleccionada. Una vez obtenida una imagen se procedió a darle un tamaño estándar, este puede ser modificado dependiendo del tamaño del monitor que se desea utilizar o de imagen que se desea dar como salida de video. Una vez obtenido el siguiente frame a procesar se suavizó la imagen utilizando un filtro de desenfoque Gaussiano (Gaussian Blur en inglés) del cual se discute más en el marco teórico, este se realizó utilizando la función `GaussianBlur()` y por último se procedió a convertir el frame al espacio de color HSV. Una vez preparada la imagen se realizó una máscara que aisló el rango de colores seleccionados, utilizando la función `InRange()` que realiza un thresholding simple de la imagen.

Una vez se tiene la imagen binaria, se procedió a utilizar la función `findContours()` para detectar los objetos que se marcaron con la máscara creada y se creó un pequeño círculo utilizando `minEnclosingCircle()` al cual se le encontró el centroide que posteriormente sería monitoreado para detectar el movimiento del mismo. Para esto se creó un delta de la posición y según esto se pudo determinar la dirección en la que el objeto se desplazó.

### 8.3. Sistema de seguimiento utilizando la Raspberry Pi 3

El propósito inicial para el sistema de seguimiento fue utilizar una Raspberry Pi 3 para controlar el sistema de forma remota, ya que puede ser programada utilizando Python y OpenCV. A gran escala se planificó el sistema como se muestra en la Figura ??, en donde la cámara está conectada directamente a la Raspberry Pi 3, esta recibe la señal de video, la procesa y envía los datos a la computadora en donde se manejan para ser distribuidos posteriormente.

Para poder utilizar la Raspberry Pi 3, fue necesario instalar una imagen de Raspian en una tarjeta SD, la versión de Python más reciente y OpenCV.

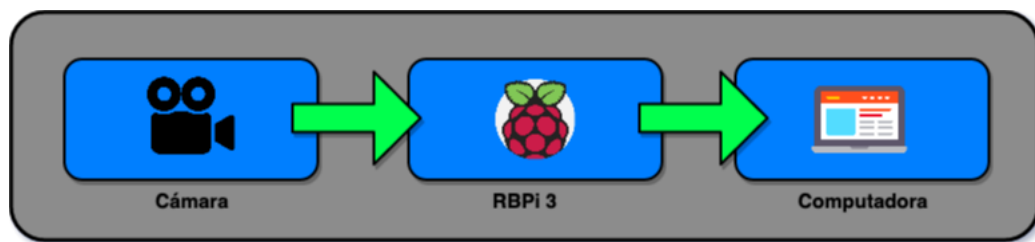


Figura 38: Sistema de seguimiento: Conexión RBPi 3

<sup>1</sup>Frame: no tiene una traducción exacta al español, pero se refiere a imágenes simples que en conjunto forman un video.

### 8.3.1. Pruebas de validación utilizando la Raspberry Pi 3

Una vez concluido el programa descrito anteriormente e instalado todo lo necesario en la Raspberry Pi 3 se procedió a realizar pruebas para verificar el funcionamiento y si se cumple con los requisitos de diseño planteados anteriormente.

La primera prueba de validación realizada fue desplegar el video tomado por la cámara web en un monitor conectado directamente en la Raspberry Pi 3, esto no tuvo ningún problema, el único cambio que se tuvo que realizar fue que por defecto se despliega el video invertido por lo que se tuvo que reflejar haciendo un pequeño cambio al programa. Con esto se pudo comprobar que el primer requerimiento se cumple, ya que fue posible conectar una cámara de video y obtener imágenes utilizando el programa diseñado.

La segunda prueba de validación realizada, luego de comprobar lo anteriormente mencionado, se basó en implementar el programa completo para identificar e imprimir en consola la dirección en que se mueve un objeto de un color establecido, para esto se utilizó un gorro de color verde brillante y se desplazó este hacia la derecha y al lado contrario y se comprobó si en consola los resultados impresos corresponden, como se muestra en la Figura 40. Esta prueba no dio ningún problema, ya que en consola se pudo desplegar correctamente la dirección en que se desplaza el objeto. El problema se dio en el momento de intentar enviar la información a la computadora para desplegar el video y los resultados obtenidos, ya que al hacer esto se detectó un lag<sup>2</sup> en la retroalimentación de video y en los datos de cambio de dirección, por lo que aunque se cumple el segundo requerimiento al intentar de implementar el tercero se pierde rendimiento y se deteriora el funcionamiento del mismo.

## 8.4. Sistema de seguimiento utilizando una computadora

Debido a los problemas que se presentaron durante las pruebas de validación y tomando en cuenta que parte del sistema final, que se desea diseñar incluye una aplicación con un retroalimentación de video, se procedió a probar el mismo programa pero directamente en la computadora.

El programa utilizado en esta etapa fue el mismo utilizado anteriormente, para esto se instaló Python en una computadora y los complementos necesarios para utilizar las librerías de OpenCV. A diferencia del sistema anterior en este caso el diagrama de bloques que describe la comunicación del sistema es como el mostrado en la Figura 39 en donde se toma la información de la cámara y envía directamente a la computadora en donde se procesa.

### 8.4.1. Pruebas de validación utilizando una computadora

Las pruebas de validación para comprobar el cumplimiento de los requerimientos fueron las mismas realizadas anteriormente, solo que con el sistema aplicado al nuevo modelo utilizando solamente la computadora, omitiendo la Raspberry Pi 3. De los resultados obtenidos

---

<sup>2</sup>Según el diccionario de Cambridge lag se refiere a un retraso, por lo que en contexto se puede definir como el retraso en una señal o de envío y recepción de información.

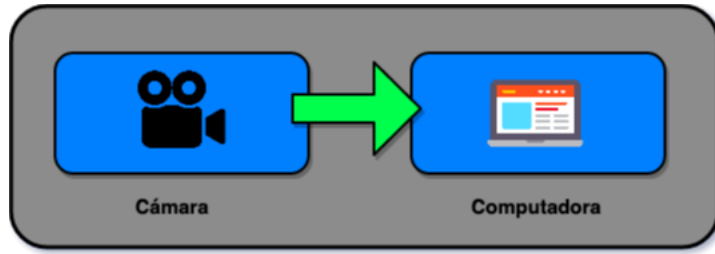


Figura 39: Sistema de seguimiento: Conexión computadora

se pudo comprobar que se cumple el primer requisito ya que se pudo obtener y desplegar imágenes por medio de la cámara de video utilizada. El segundo requisito también se cumplió ya que se logró obtener, por medio del programa, datos suficientes para determinar la dirección de desplazamiento del objeto y como se muestra en la Figura 40 se pudo desplegar un feed de video, por lo que se puede concluir con que se cumplen los tres requisitos planteados para el diseño del seguimiento mediante una cámara de video.

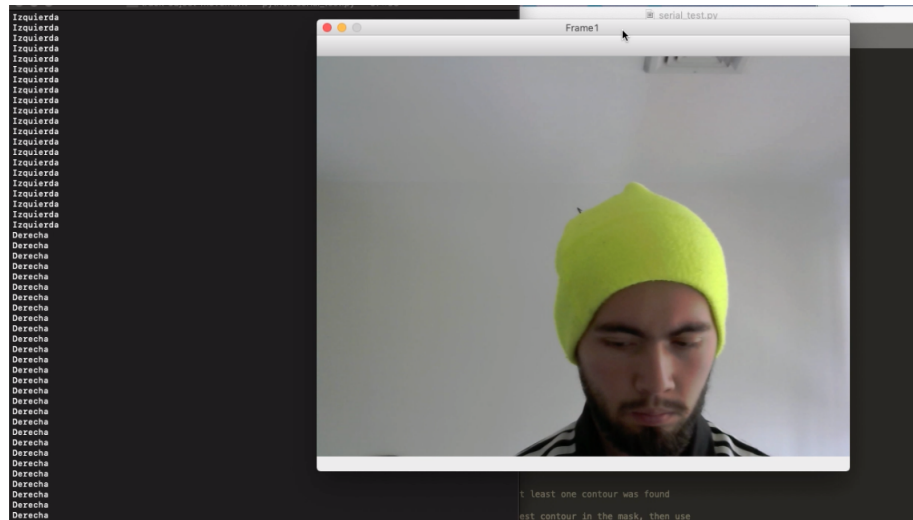


Figura 40: Sistema de seguimiento: Prueba de seguimiento.

## 9.1. Requisitos del sistema

Adicional al sistema de seguimiento desarrollado anteriormente, es parte fundamental del sistema que este programa pueda comunicarse correctamente con un motor y dictarle la dirección en que debe girar para poder desplazar al sistema. Por esta razón se añade estos requerimientos al sistema de seguimiento planteado anteriormente.

- Se debe poder enviar la información obtenida por el sistema, a un microcontrolador.
- El microcontrolador debe obtener la información e interpretarla correctamente.
- El microcontrolador debe enviar una señal de control al motor para accionarlo correctamente.

## 9.2. Diseño de sistema

Se continuó con el diagrama de bloques utilizado en el capítulo anterior y se le agregó el microcontrolador y motor, como se muestra en la Figura 41, en donde de la computadora se envía la señal al controlador y de este al circuito que controla el motor.

Como controlador para el sistema se utilizó el Arduino UNO, ya que su entorno de programación permite muchas facilidades para realizar cambios inmediatos y se comunicó con la computadora (Python) por medio de comunicación serial a través del puerto USB de la computadora a una razón de 9600 Baudios.

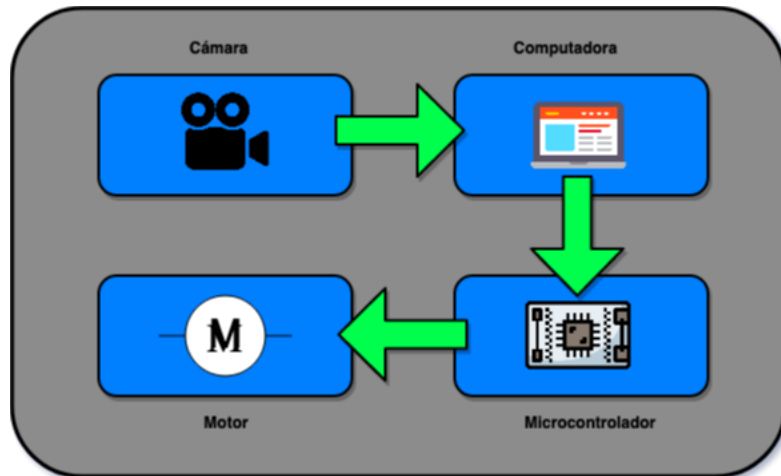


Figura 41: Implementación de motor: Prueba de seguimiento.

Para enviar la señal desde Python hacia el Arduino se utilizan caracteres codificados, esto quiere decir que se envía un carácter que tiene un significado que el Arduino interpreta y según esto toma una decisión de acción. En la Tabla 6 se muestra los caracteres enviados desde la computadora y la interpretación que se les da en el Arduino. Estos caracteres se envían con cada actualización de frame de la cámara y en el momento en que se deja de obtener una imagen de la misma. El Arduino espera si hay información en el puerto serial, en el momento en que hay información en la cola este la lee y la almacena en una variable. Esta variable temporal que estará siendo actualizada constantemente se compara por medio de condicionales para entrar en distintas partes del programa en donde se envían las señales para activar el motor.

Carácter	Descripción
'a'	El objeto se mueve a la derecha.
'b'	El objeto se mueve a la izquierda.
'c'	El objeto NO se mueve.
'q'	Se ha salido del programa.

Cuadro 6: Implementación de motor: Caracteres enviados.

Los motores utilizados están conectados a las salidas de un LN298 mini, el cual recibe señales del Arduino y dependiendo de estas cambia la polaridad de la señal a los motores para decidir la dirección en que estos giran y en caso todos los pines estén apagados (LOW) el motor se detiene. Algo importante que se implementó desde el Arduino es el control, por medio de una señal PWM, de las velocidades de arranque y en de en la que se detiene el motor, en caso el motor cambie de dirección o se quiera detener este se detiene lentamente cambiando la frecuencia de la señal de entrada y del mismo modo pero a la inversa este acelera hasta llegar a su velocidad máxima.

### 9.3. Pruebas de validación

Basado en los requisitos que el sistema debe de cumplir se procedió a comprobar el envío de datos desde la aplicación, para realizar esto se imitó la prueba realizada en el capítulo anterior, en donde se conectó la cámara y con un gorrito de color verde brillante se movió lateralmente, con la única diferencia de que esta vez a parte de imprimir en consola, para verificar, se envió los datos y se comprobó utilizando un monitor serial. Una vez se comprobó que esto funciona correctamente se procedió a verificar conectando los motores al driver y este al controlador, y con la misma prueba realizada anteriormente se comprobó que los motores cambian de dirección consistentemente al cambiar la dirección de movimiento del gorro. Con esto se puede concluir que se cumplen todos los requisitos planteados para la implementación de un motor al sistema, como se puede observar en el siguiente [video](#).

---

## Selección de motor

---

Para la implementación del prototipo se utilizó dos motores DC con un voltaje máximo de 6 Voltios y una velocidad aproximada de 130 RPM. Es evidente que si el sistema se va a utilizar para dar seguimiento a un nadador, este debe de poder trasladar el carrito a una velocidad similar. Por esta razón una vez terminada la construcción de la parte mecánica del prototipo se hicieron pruebas de velocidad, para estas pruebas se grabó un video en el cual se observa al sistema desplazarse de un punto a otro (conocidos) para posteriormente verificar el tiempo que esto le tomaba y de esta forma verificar si las velocidades coinciden con las del fabricante. Según el fabricante el motor gira a una velocidad de 130 RPM, pero como lo que se obtendría y con lo que se desea comparar, es una velocidad lineal se procedió a realizar una conversión tomando en cuenta que el diámetro del eje del motor (incluyendo al piñón utilizado) es de 8 milímetros utilizando el siguiente modelo:

$$v = \frac{\omega * 2\pi * r}{60} \quad (4)$$

En donde  $\omega$  es la velocidad angular del motor y  $r$  es el radio del eje (incluyendo al piñón utilizado). Con esto se obtuvo que la velocidad lineal es de aproximadamente 0.108 m/s. Si se observa la Tabla 7 se puede ver que a una velocidad con los motores alimentados con 3 Voltios, el porcentaje de error entre la velocidad que proporciona el fabricante y la obtenida en las pruebas es aproximadamente en promedio de 17%, mientras que si se observa en la Tabla 8 en donde el voltaje se aumenta a 5 Voltios, los porcentajes de error aumentan bastante. De esto se puede concluir que el fabricante brinda una velocidad media y no una velocidad máxima.

Una vez obtenidas estas velocidades se puede proceder a realizar comparaciones con las velocidades de los records obtenidos por los nadadores olímpicos (mencionados en el marco teórico). Actualmente uno de los nadadores olímpicos que tiene unos de los records

más rápidos es Michael Phelps con una velocidad aproximada de 2.7 m/s, utilizando esta como un máximo para un marco de comparación se puede concluir rápidamente que los motores utilizados no pueden proporcionar una velocidad suficiente para seguir a un nadador mientras compite. Por esta razón se procedió a diseñar en base a esta velocidad para poder luego buscar un motor que pueda cumplir con este requisito.

Prueba	Velocidad	Velocidad teórica	Error
1	0.10 m/s	0.108 m/s	8.2 %
2	0.08 m/s	0.108 m/s	26 %
<b>Promedio</b>	0,09 m/s	0.108 m/s	17 %

Cuadro 7: Pruebas de velocidad motor 3V

Prueba	Velocidad	Velocidad teórica	Error
1	0.133 m/s	0.108 m/s	22 %
2	0.20 m/s	0.108 m/s	83 %
3	0.20 m/s	0.108 m/s	83 %
<b>Promedio</b>	0.178	0.108 m/s	63 %

Cuadro 8: Selección de motor: Pruebas de Velocidad 5V

Para diseñar para un motor que cumpla con las características de velocidad es necesario, a partir de la velocidad lineal, encontrar una velocidad angular, para esto se procedió a calcular por medio del modelo utilizado anteriormente, pero despejado para la velocidad angular  $\omega$ :

$$\omega = \frac{60v}{2\pi r} \quad (5)$$

Y variando el radio del eje del motor (con el piñón incluido) se procedió a iterar hasta obtener un valor de velocidad angular que se considere adecuado. Con un radio de eje como el que se utiliza actualmente (8 milímetros) el valor de velocidad angular necesario en el motor es de aproximadamente 32,000 RPM como se muestra en la Tabla 9, lo cual se considera un poco elevando. Para la siguiente iteración se aumentó el radio (20 milímetros) y al realizar el cálculo se obtuvo los resultados mostrados en la Tabla 10, como se puede observar el valor de la velocidad angular disminuyó por más de la mitad y a pesar de que este valor es más sencillo de conseguir a nivel comercial se realizó una iteración más (radio de 50 milímetros) obteniendo un motor con una velocidad angular aproximada de 5100 RPM, como se muestra en la Tabla 11.

<b>Velocidad deseada</b>	2.7 m/s
<b>Radio Piñó (Eje)</b>	0.008
<b>RPM</b>	31809

Cuadro 9: Selección de motor (Radio = 0.008)

Con estos valores obtenidos se procedió a buscar en Digi-Key<sup>1</sup> motores que cumplan con estas especificaciones de velocidad. Los motores encontrados se muestran en la Tabla 12 Para

<sup>1</sup><https://www.digikey.com> es una tienda online de componentes electrónicos.

<b>Velocidad deseada</b>	2.7 m/s
<b>Radio Piñón (Eje)</b>	0.02
<b>RPM</b>	12723

Cuadro 10: Selección de motor (Radio = 0.02)

<b>Velocidad deseada</b>	2.7 m/s
<b>Radio Piñón (Eje)</b>	0.05
<b>RPM</b>	5089

Cuadro 11: Selección de motor (Radio = 0.05)

cada uno de los motores seleccionados se listó el link a la página en donde se pueden comprar y el enlace a la hoja de datos para verificar sus especificaciones, esto se puede encontrar en la Tabla [13](#).

<b>Nombre</b>	<b>Fabricante</b>	<b>Voltaje máximo</b>	<b>RPM max.</b>	<b>Torque max.</b>
M1N10FB11G	NMB Technologies Corporation	5	15906	2,25
SE8G0NTM	NMB Technologies Corporation	5	12500	0,88
13N88-216E.1	Portecap	5	10800	3.2
13N88-110.1	Portecap	12	12400	3,33
17n78210E.1	Portecap	12	10000	4,8

Cuadro 12: Motores evaluados

Como se puede observar se eligió motores con torques máximos distintos, esto con el objetivo de que una vez se tenga construido un prototipo final con materiales impermeables, se pueda pesar y calcular cuál sería el motor ideal para colocar al sistema.

<b>Nombre</b>	<b>Hoja de datos</b>	<b>Página</b>
M1N10FB11G	<a href="#">M1N10FB11G</a>	<a href="#">M1N10FB11G</a>
SE8G0NTM	<a href="#">SE8G0NTM</a>	<a href="#">SE8G0NTM</a>
13N88-216E.1	<a href="#">13N88-216E.1</a>	<a href="#">13N88-216E.1</a>
13N88-110.1	<a href="#">13N88-110.1</a>	<a href="#">13N88-110.1</a>
17n78210E.1	<a href="#">17n78210E.1</a>	<a href="#">17n78210E.1</a>

Cuadro 13: Motores evaluados, enlaces a página y hoja de datos.

## 11.1. Requisitos de la interfaz gráfica

Para el diseño de la interfaz de usuario se planteó tres requisitos que se consideran fundamentales para el correcto funcionamiento de la misma para la aplicación que se desea darle.

- La interfaz de usuario debe permitir mostrar un feed de video.
- La interfaz de usuario debe permitir que el usuario guarde y acceda a videos anteriores.
- La interfaz de usuario debe permitir iniciar una nueva rutina.

## 11.2. Diseño de la interfaz gráfica

Para el diseño de la interfaz gráfica se consideró desde un inicio continuar utilizando Python, ya que este es el lenguaje de programación que se utilizó para el sistema de seguimiento del proyecto. Por esta razón se procedió a utilizar PyQt, que como se explica más a detalle en el marco teórico, permite realizar interfaces gráficas de usuario y por medio de uno de sus módulos convertir el código al lenguaje de Python. Inicialmente se planteó un mapa de funcionamiento de la interfaz que se puede observar en la Figura 42, en base a esto se procedió a diseñar la parte gráfica utilizando la aplicación de Qt Design, en esta se colocaron todos los widgets de una forma en la que se viera visualmente agradable y se pudiera acceder a cada una de las ramas de la aplicación de manera sencilla. En la Figura 43 se muestra una captura del diseño preliminar de esta, en donde aún no se han aplicado temas ni imágenes.

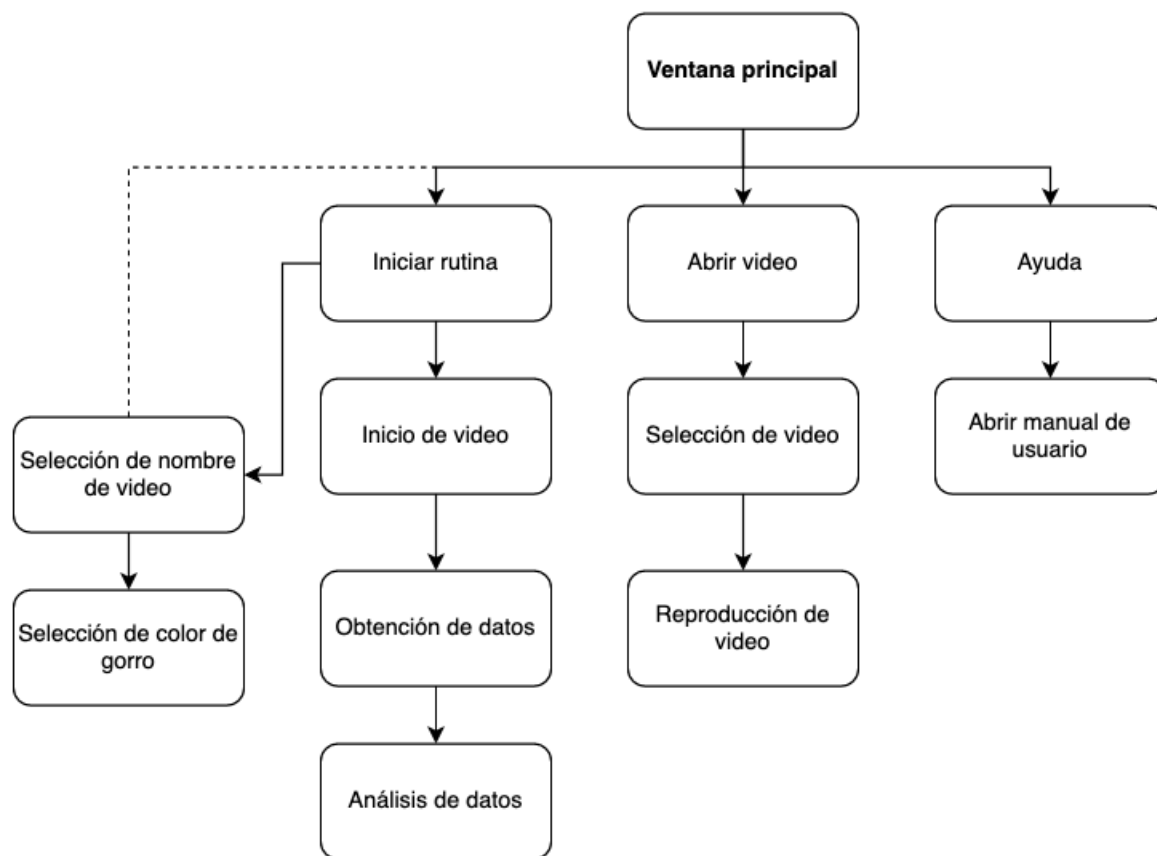


Figura 42: Interfaz de Usuario: Mapa de la aplicación

Una vez terminado el diseño en QtDesigner, se procedió a exportar el código a un archivo .ui, que es el tipo de archivo que genera este programa. Para trasladar este código al lenguaje de programación de Python se utilizó un comando en la consola, es importante dirigirse a la carpeta en la que se encuentra el archivo .ui para que esto funcione, el archivo .py será creado en esta misma carpeta.

El comando utilizado es el siguiente:

```
$ cd dirección de archivo.ui #Dirigir a la carpeta donde se encuentra el archivo.
```

```
$ pyuic5 -x archivo.ui > archivo.py # Transformar archivo.
```

Con esto se obtuvo un código en el lenguaje de Python que cuenta con una clase que es la interfaz gráfica, a continuación se procedió a realizar una unión de código entre lo que se tenía y la interfaz. Para esto se creó distintas funciones las cuales se aliaron a una acción dentro de la interfaz para que estas entraran en funcionamiento en el momento en que el usuario decidiera hacer algo. Una vez terminado esto se agregó algunas funcionalidades útiles para mejorar la experiencia, como seleccionar el color de gorro que desea utilizar el atleta y nombrar el video que se va a generar en la sesión. (Adicional al nombre se agrega la fecha y hora en que se realiza la sesión de práctica). En la Figura 44 se puede observar el resultado

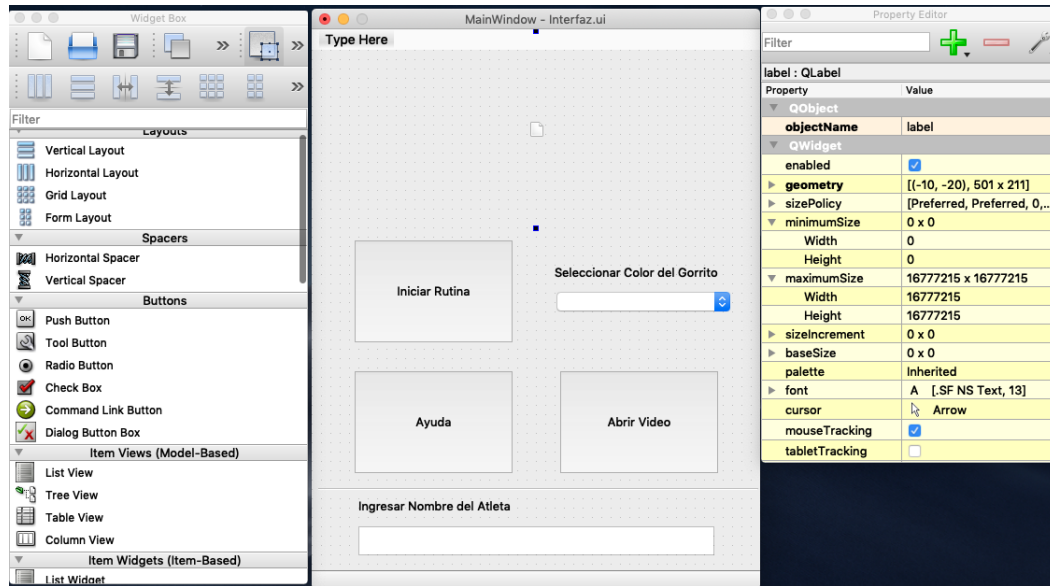


Figura 43: Interfaz de Usuario: Diseño

final de la interfaz gráfica.

### 11.3. Pruebas de validación

Para validar que la interfaz cumple con los requisitos planteados en un inicio se procedió a probar el sistema completo, incluyendo el prototipo mecánico y el sistema de seguimiento. Una vez realizadas las conexiones del sistema se abrió la aplicación, al presionar el botón para iniciar una nueva rutina, colocando previamente el color de gorro a utilizar, se logró realizar el seguimiento de una bufanda de color verde brillante como se muestra en la Figura 45 y posteriormente una bufanda de color rojo, también se logró crear un nuevo video con la fecha y la hora del inicio de la prueba como nombre ya que se dejó en blanco el espacio de nombre del video, lo cual por defecto crea un video con estas características. Durante todo el funcionamiento del sistema se tuvo un feed de video, como el que se muestra en la Figura 46, donde se evidencia que la bufanda está dentro del campo de visión de la cámara que lo seguía.

Posteriormente, desde la aplicación se ingresó a buscar los videos generados durante las pruebas mencionadas anteriormente, al abrir el cuadro de texto se seleccionó uno de los videos creados, como se muestra en la Figura 48. Con esto último, se pudo comprobar que se cumple con los requisitos planteados en un inicio, por lo que se puede concluir que la interfaz de usuario es funcional para este sistema.

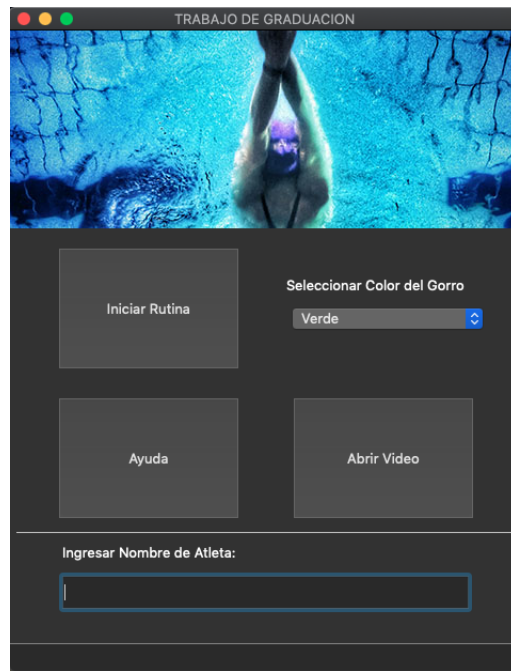


Figura 44: Interfaz de Usuario: Resultado final

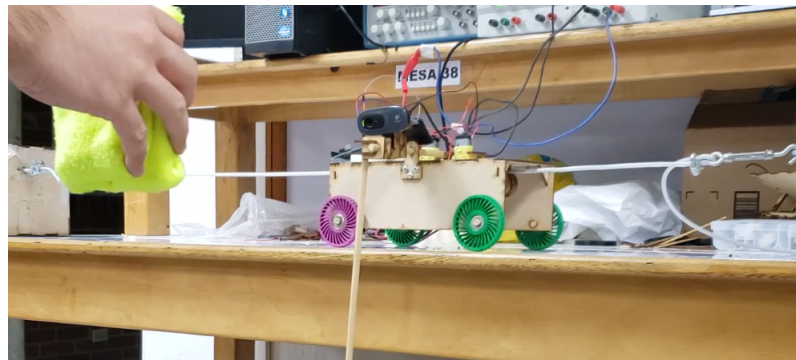


Figura 45: Interfaz de Usuario: Prueba con prototipo



Figura 46: Interfaz de Usuario: Prueba con prototipo desde cámara de video.

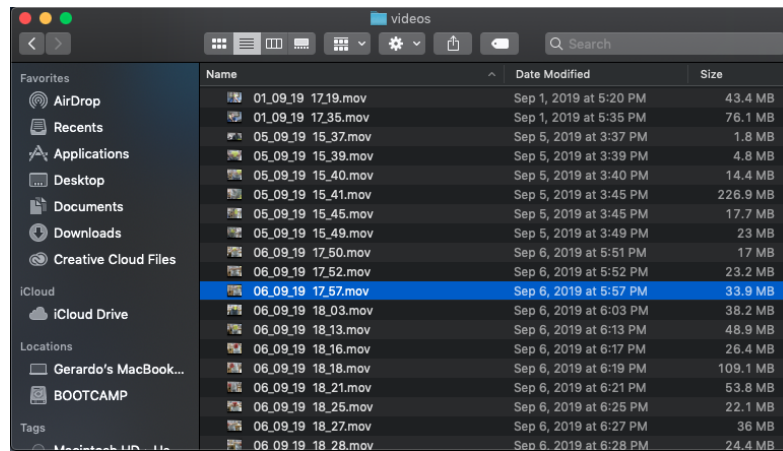


Figura 47: Interfaz de Usuario: Generar videos.

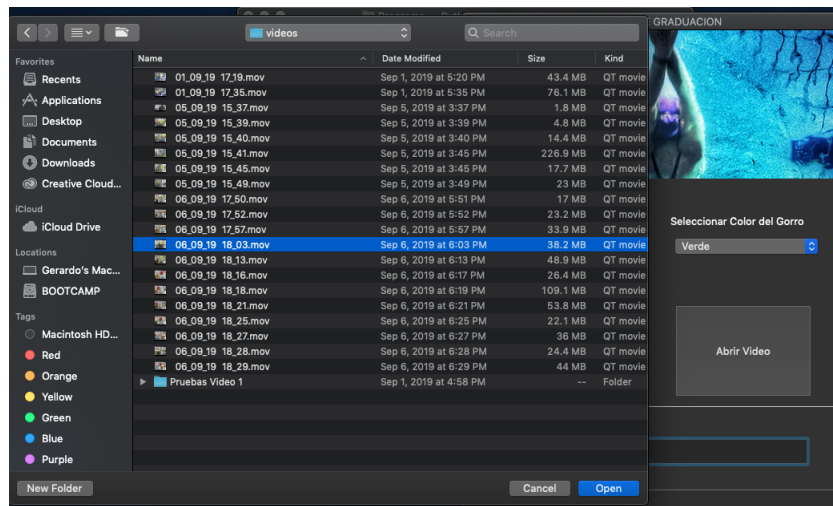


Figura 48: Interfaz de Usuario: Abrir videos.

Se espera que en un futuro el prototipo manufacturado pueda ser construido con las especificaciones necesarias para poder funcionar en un ambiente húmedo y propenso a salpicaduras de agua, por esta razón se realizó una pequeña investigación de algunos materiales que se deben de considerar para construir un prototipo final, adicionales a algunos que ya se han utilizado.

### 12.1. Requisitos de los materiales

Se seleccionó algunos requisitos que se considera, los materiales utilizados para la construcción de un prototipo futuro deben de tomarse en cuenta.

- Los materiales seleccionado para la fabricación del dispositivo deben soportar salpicaduras de agua y permitir estar en un ambiente posiblemente húmedo.
- Los materiales a utilizar deben permitir un proceso de manufactura sencillo.
- Los materiales deben ser de fácil acceso comercial.

### 12.2. Cable de acero con alma de yute

El cable de acero es una máquina compuesta de un número de capas movibles, diseñado y manufacturado para ofrecer rigidez, precisión en movimientos simples con una función definitiva. Este cable está formado por tres componentes básicos que se muestran en la Figura 49 a continuación, en donde se puede observar el alma, el alambre y el torón. 25

- **Alambre:** la principal materia prima del alambre es acero, este puede ser de distintos calibres y en distintos grados, dependiendo de la aplicación.
- **Torón:** el torón es formado en al entrelazar o enrollar varios alambres cubriendo un elemento central, el alma en este caso. Nuevamente en la Figura 49 se puede observar este procedimiento en el cable de acero.
- **Alma:** el alma es la parte central del cable de acero, este es el elemento en el cual se enrollan los elementos para formar el torón. Su principal función es brindar soporte a la hora de operar.

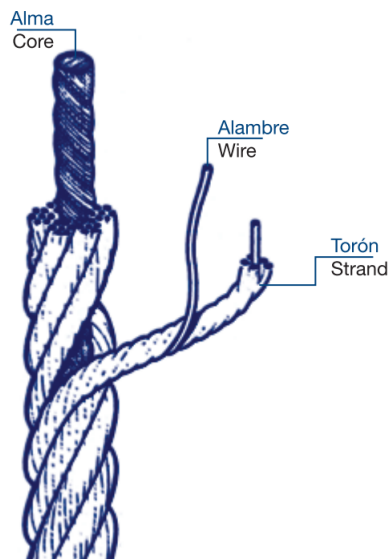


Figura 49: Materiales adicionales: Cable de acero

25

Una de las ventajas que se puede destacar del cable con Alma de Yute, es que comercialmente este se puede conseguir con un recubrimiento plástico, lo cual le permite estar expuesto a condiciones externas sin que este se oxide o dañe.

### 12.3. Perfiles de aluminio

El Aluminio es un metal no ferroso plateado muy ligero, resistente a temperaturas de hasta 660°C, su densidad relativa es de 2.7 kg/m<sup>3</sup>. Este material es uno de los de mayor consumo a nivel mundial y es el metal más abundante en la corteza terrestre. 26 Una de las características más importantes de este material es su buen comportamiento a la corrosión, esto debido a la presencia permanente de un óxido natural. A pesar de que el grosor de esta capa de óxido es microscópica lo hace pasivo a la acción del medio ambiente, esta pequeña capa se forma inmediatamente al tener contacto con un medio oxidante en el medio ambiente, que puede ser el oxígeno, el aire o el agua. 27 Otras características importantes pueden ser su conductividad eléctrica, la cual lo hace un rival digno del cobre pero siendo este tres

veces menos pesado por lo que puede llegar a ser mucho más económico; otra característica importante es su buena conducción térmica, no es magnético ni tóxico para el ser humano. [26]

Una característica fundamental del aluminio es que es un material considerado muy bueno para los trabajos de maquinado mecánico, este material es el más utilizado en formas de perfiles ya que tiene una buena resistencia mecánica. [28]

### 12.3.1. Resistencia al agua y a la humedad

El agua es un agente que puede causar un daño superficial al aluminio, esta puede causar un óxido en la cara del material si no es tratado con cuidado. Por lo general, las manchas que suelen aparecer en este material son a causa de salpicaduras de agua que no fueron retiradas en un tiempo, por lo que se recomienda que si este tiene contacto con agua se seque antes de ser guardado. [26]

La humedad en sí, como un componente ambiental, no es capaz de dañar el aluminio ya que este soporta la acción de este tipo de oxidantes. A pesar de esto la humedad sí puede ser un accionador de óxido hacia el aluminio, ya que puede disolver uno o varios de los constituyentes de un producto químico y dependiendo de la naturaleza de estos productos el resultado puede o no ser agresiva y producir una corrosión que cause picaduras o una corrosión uniforme en el material. [27]

## 12.4. Cable resistente al agua

Si se desea construir el sistema cercano a una piscina, es necesario que los cables de alimentación del sistema, de los motores y de los circuitos estén protegidos para evitar problemas. Existen muchos tipos de cables conductores eléctricos, algunos de estos son:

- **Conductor de alambre desnudo:** es un alambre, sólido, sin recubrimiento de ningún tipo.
- **Conductor de alambre aislado:** es lo mismo que un conductor de alambre desnudo, con la diferencia de que tiene una cobertura de material aislante plástico.
- **Conductor de cable flexible:** este tipo de cable es el más comercial y está compuesto de varios alambres delgados, recubiertos por una cobertura aislante plástica. La ventaja más grande de este tipo de cable es su maleabilidad.
- **Conductor de cordón:** estos están formados por más de un cable o alambre, se juntan todos y se juntan con otra capa de aislante plástico que los cubre a todos.

Los cables pueden tener varios tipos de aislamiento, por lo general se puede saber qué tipo de aislamiento tiene si se lee una pequeña inscripción en la cobertura con abreviaciones en inglés. [29]

- **T(Termoplástico):** es un aislamiento termoplástico que se encuentra en todos los cables.
- **H(Resistente al Calor):** es resistente al calor (hasta los 75°).
- **HH(Resistente al Calor):** es resistente al calor (hasta los 90°).
- **W(Resistente al Agua):** resistente al agua y la humedad.
- **LS(Baja Emisión de Humo):** este cable tiene una baja emisión de humo o gases.

- Se diseñó y construyó el prototipo de un sistema que se desliza horizontalmente y permite montar en él una cámara de video.
- Se comprobó que el sistema, alimentado con 5 Voltios, puede desplazarse a una velocidad máxima de 0.20 m/s en una longitud de desplazamiento de aproximadamente 1 metro.
- Se determinó que al estar el sistema expuesto a salpicaduras de agua y a humedad este debe de ser construido con materiales como el aluminio, que con un buen cuidado, puede soportar estas condiciones, cable de acero con alma de yute con un recubrimiento plástico, cables debidamente seleccionados y polímeros que se puedan utilizar en impresoras 3D como lo es el TPU.
- Se controló el sistema, obteniendo datos directamente de una cámara de video a una computadora y por medio de protocolo serial, se envió datos al microcontrolador para el manejo de los motores.
- Se programó una interfaz gráfica que permite al usuario controlar el sistema mecánico, obtener una retroalimentación de video inmediata, seleccionar el color del gorro a utilizar por el atleta y almacenar los videos grabados durante las sesiones y abrirlos para reproducirlos posteriormente.
- Se comprobó que el sistema es capaz de dar seguimiento a objetos de color rojo, que se encuentran en el rango del modelo HSV que comprende entre  $HSV_{\min}(0,50,50)$  y  $HSV_{\max}(10,255,255)$ ; objetos de color verde, que se encuentran en el rango del modelo HSV que comprende entre  $HSV_{\min}(29,86,6)$  y  $HSV_{\max}(64,255,255)$  y de color azul, que se encuentran en el rango del modelo HSV que comprende entre  $HSV_{\min}(216, 100,42)$  y  $HSV_{\max}(222,67,30)$ .
- Se desarrolló un manual de usuario que muestra cómo el sistema puede ser ensamblado, montado y utilizado sencillamente siguiendo una serie de pasos.

A personas interesadas en este proyecto para un futuro se les proporciona algunas recomendaciones que podrían ser de utilidad al retomar el proyecto.

1. Para alcanzar una velocidad similar a la de los atletas en la natación, utilizar uno de los motores mencionados en el capítulo "Diseño de motores".
2. Maquinar y construir el sistema a gran escala y tomar en cuenta los materiales mencionados en el capítulo de "Materiales adicionales"
3. Para implementar el sistema en una piscina, consultar al COG si cuenta con una cámara para poder adaptar al sistema.
4. Comprobar el funcionamiento utilizando una pantalla portátil conectada de forma remota (sea por medio de un cable o inalámbrica) a la Raspberry Pi, que estaría montada en el carrito para desplegar la interfaz de usuario y evitar el uso de una computadora.

- [1] J. Pers y S. Kovacic, “Computer vision system for tracking players in sports games”, en *IWISPA 2000. Proceedings of the First International Workshop on Image and Signal Processing and Analysis. in conjunction with 22nd International Conference on Information Technology Interfaces.* (IEEE, IEEE, 2000, págs. 177-182.
- [2] E. Pogalin, A. H. Thean, J. Baan, N. Schipper, A. W. Smeulders y col., “Video-based training registration for swimmers”, *International Journal of Computer Science in Sport*, vol. 6, n.º 1, págs. 4-17, 2007.
- [3] F. A. Magalhaes, Z. Sawacha, R. Di Michele, M. Cortesi, G. Gatta y S. Fantozzi, “Effectiveness of an automatic tracking software in underwater motion analysis”, *Journal of sports science & medicine*, vol. 12, n.º 4, pág. 660, 2013.
- [4] J. S. Perea y A. H. Castro, *Técnicas específicas de nado en el medio acuático: certificado de profesionalidad*. MAD, 2012.
- [5] M. G. C. José, *Enseñanza de la natación: salidas, llegadas y virajes: biomecánica, barrios, acción de piernas: Manual para monitores, profesores y practicantes*. Wanceulen, 2012.
- [6] B. Lucero, *Technique swim workouts*. Meyer Meyer Sport, 2009.
- [7] *Swimming Equipment and History - Olympic Sport History*, dic. de 2018. dirección: <https://www.olympic.org/swimming-equipment-and-history>.
- [8] Ioc, *Olympic Games: Swimming*, jul. de 2019. dirección: <https://www.olympic.org/swimming>.
- [9] *Current Olympic Record Holders in Swimming: Olympics at Sports*. dirección: <https://www.sports-reference.com/olympics/friv/lists.cgi?id=68>.
- [10] M. A. A. S. NIXON, *FEATURE EXTRACTION AND IMAGE PROCESSING FOR COMPUTER VISION*. ELSEVIER ACADEMIC PRESS, 2019.
- [11] *What is PyQt?* Dirección: <https://riverbankcomputing.com/software/pyqt/intro>.
- [12] G. Bradsky y A. Kaehler, *Learning Open CV Computer vision with the Open CV library*. OReilly, 2008.

- [13] J. J. Esqueda, *Fundamentos de Procesamiento de Imágenes*. UABC, 2005.
- [14] *What Is Image Filtering in the Spatial Domain?* Dirección: <https://www.mathworks.com/help/images/what-is-image-filtering-in-the-spatial-domain.html>.
- [15] *Image Filtering*. dirección: <https://docs.opencv.org/2.4/modules/imgproc/doc/filtering.html#blur>.
- [16] Dirección: [https://pippin.gimp.org/image-processing/chap\\_area.html](https://pippin.gimp.org/image-processing/chap_area.html).
- [17] *IMAGE PROCESSING AND COMMUNICATIONS: techniques, algorithms and applications*. SPRINGER NATURE, 2019.
- [18] *High-dimensional Gaussian Filtering for Computational Photography*. dirección: <https://books.google.com.gt/books?id=ihe0nfm97eIC&printsec=frontcover&dq=image+filtering+Gaussian+Blur&hl=es&sa=X&ved=0ahUKEwiGsezo2M7kAhWCzlkKHS5FA-sQ6AEIJjAA#v=onepage&q=image%20filtering%20Gaussian%20Blur&f=false>.
- [19] M. Sonka, V. Hlavac, R. Boyle, J. Xing y H. Ai, *Tu xiang chu li, Fen xi yu ji qi shi jue = Image processing, analysis, and machine vision*. Qing hua da xue chu ban she, 2016.
- [20] M. Petrou y C. Petrou, *Image processing: the fundamentals*. John Wiley y Sons, 2015.
- [21] P. Corke, *Robotics, vision and control - fundamental algorithms in matlab (r) second*, Springer International Publish, 2017.
- [22] M. McGrath, *Raspberry Pi 3 in easy steps*. In Easy Steps Limited, 2016.
- [23] A. Pajankar y J. Lentin, *Raspberry Pi Image Processing Programming: Develop Real-life Examples with Python, Pillow, and Scipy*.
- [24] *RigWheels MWS3 MicroWheel Stage-3 Camera Slider Carriage*. dirección: [https://www.bhphotovideo.com/c/product/1037644-REG/rigwheels\\_mws3\\_microwheel\\_stage\\_3\\_camera.html](https://www.bhphotovideo.com/c/product/1037644-REG/rigwheels_mws3_microwheel_stage_3_camera.html).
- [25] *Cable de Acero Catálogo*. 2008, pág. 40.
- [26] *Ficha Técnica del Aluminio*. dirección: <https://www.madecentro.com/media/ficha-tecnica/aluminio.pdf>.
- [27] *El Aluminio*. dirección: <https://www.alu-stock.es/multimedia/descargas/13/Cap10-el-aluminio.pdf>.
- [28] *Maquinado del Alumino*. dirección: [http://catarina.udlap.mx/u\\_dl\\_a/tales/documentos/lim/gama\\_o\\_je/capitulo2.pdf](http://catarina.udlap.mx/u_dl_a/tales/documentos/lim/gama_o_je/capitulo2.pdf).
- [29] *¿Que tipos de cables eléctricos existen?* Dirección: <https://masvoltaje.com/blog/tipos-de-cables-electricos-que-existen-n12>.

### 16.1. ANEXO A

Enlace al repositorio con el programa final: [REPOSITORIO](#)

### 16.2. ANEXO B

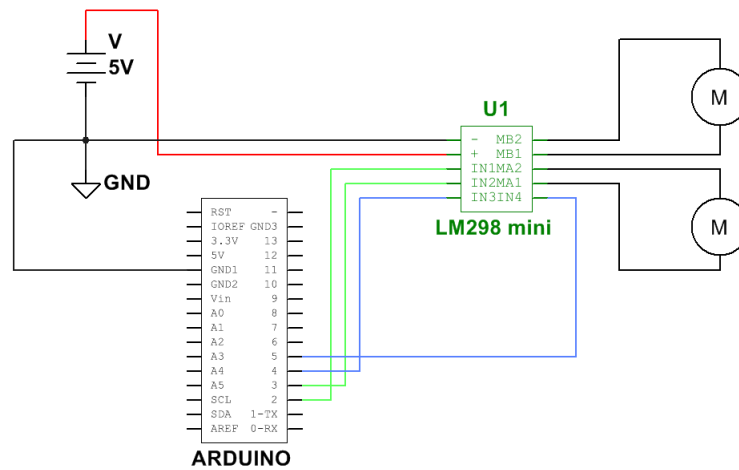


Figura 50: Circuito para conexión de motores

## 16.3. ANEXO C

### 16.3.1. ANEXO C.1

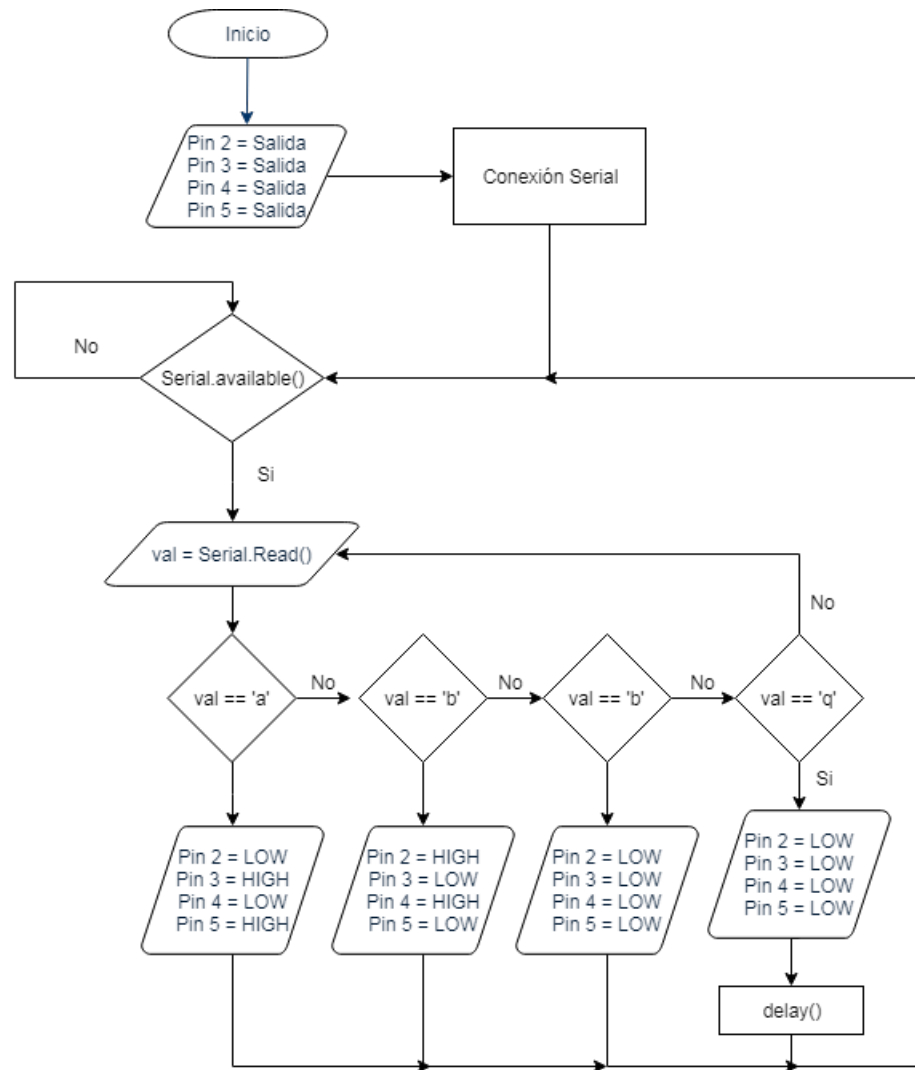


Figura 51: Diagrama de flujo de programa en Arduino

16.3.2. ANEXO C.2

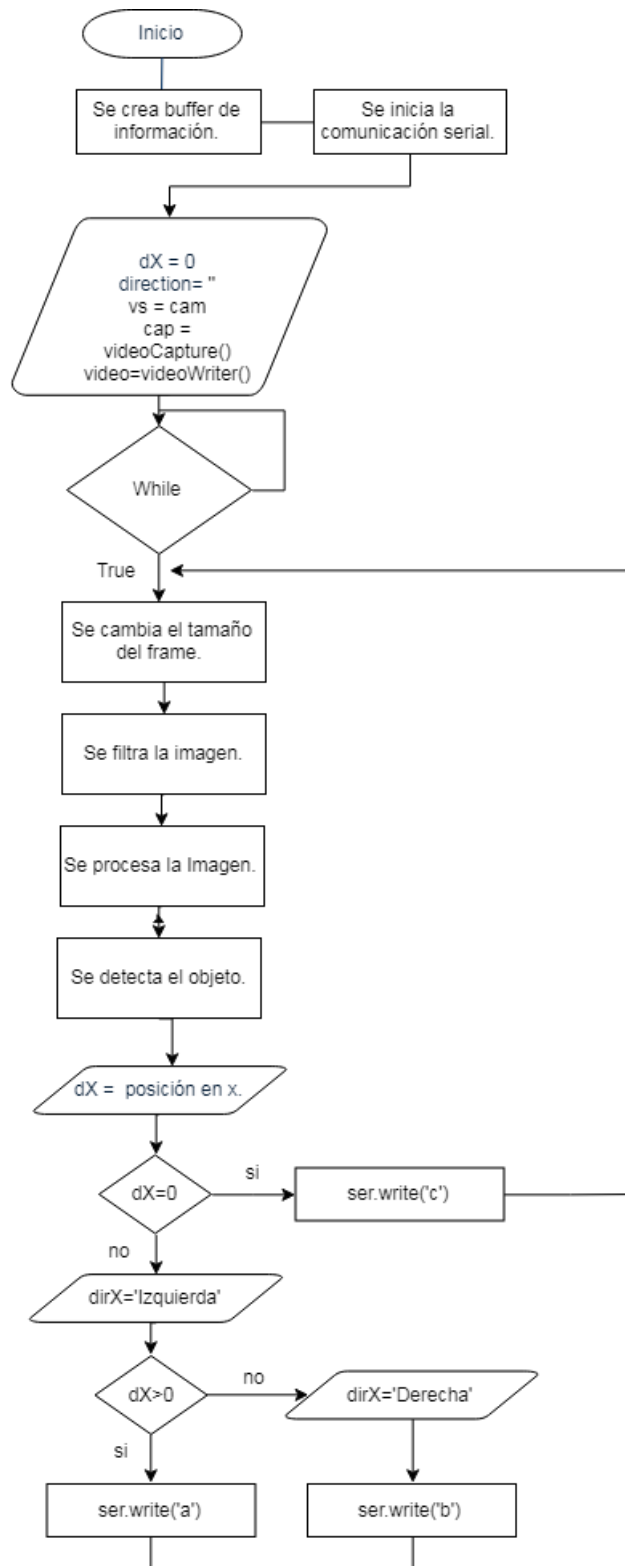


Figura 52: Circuito para conexión de motores

## 16.4. ANEXO D

Planos de interés.





16.4.3. ANEXO D.3

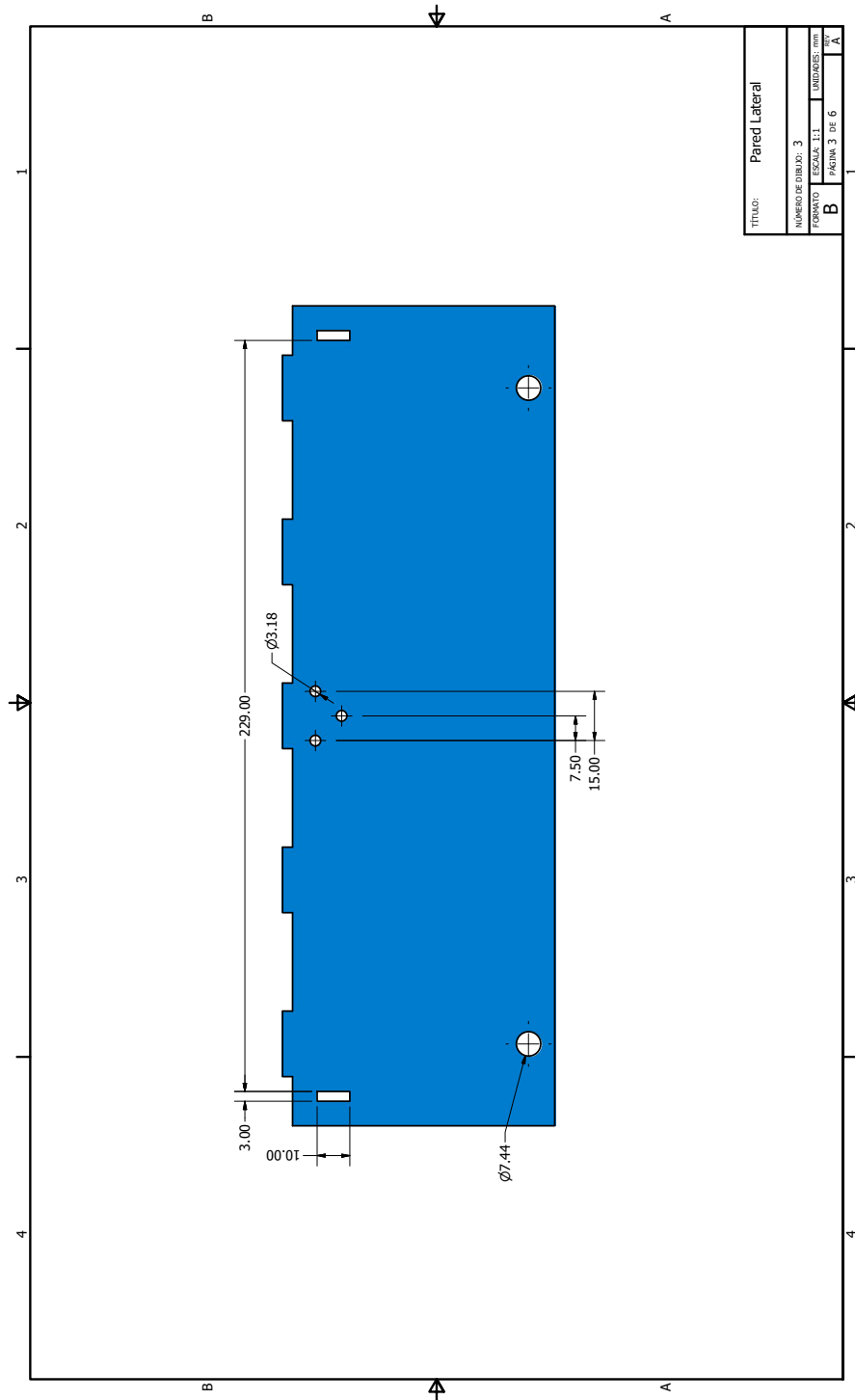


Figura 55: Plano del sistema completo Hoja 3

16.4.4. ANEXO D.4

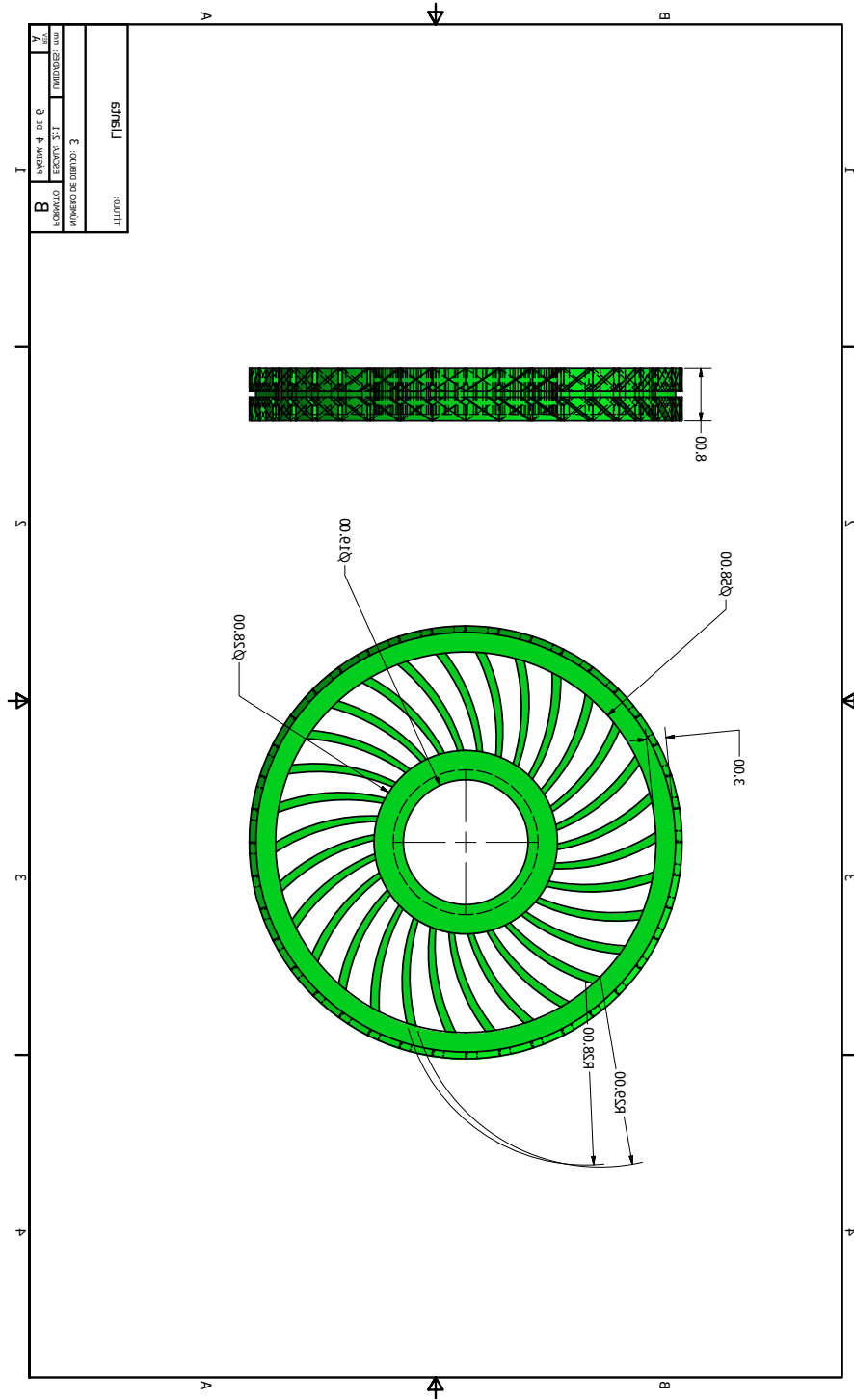


Figura 56: Plano del sistema completo Hoja 4

16.4.5. ANEXO D.5

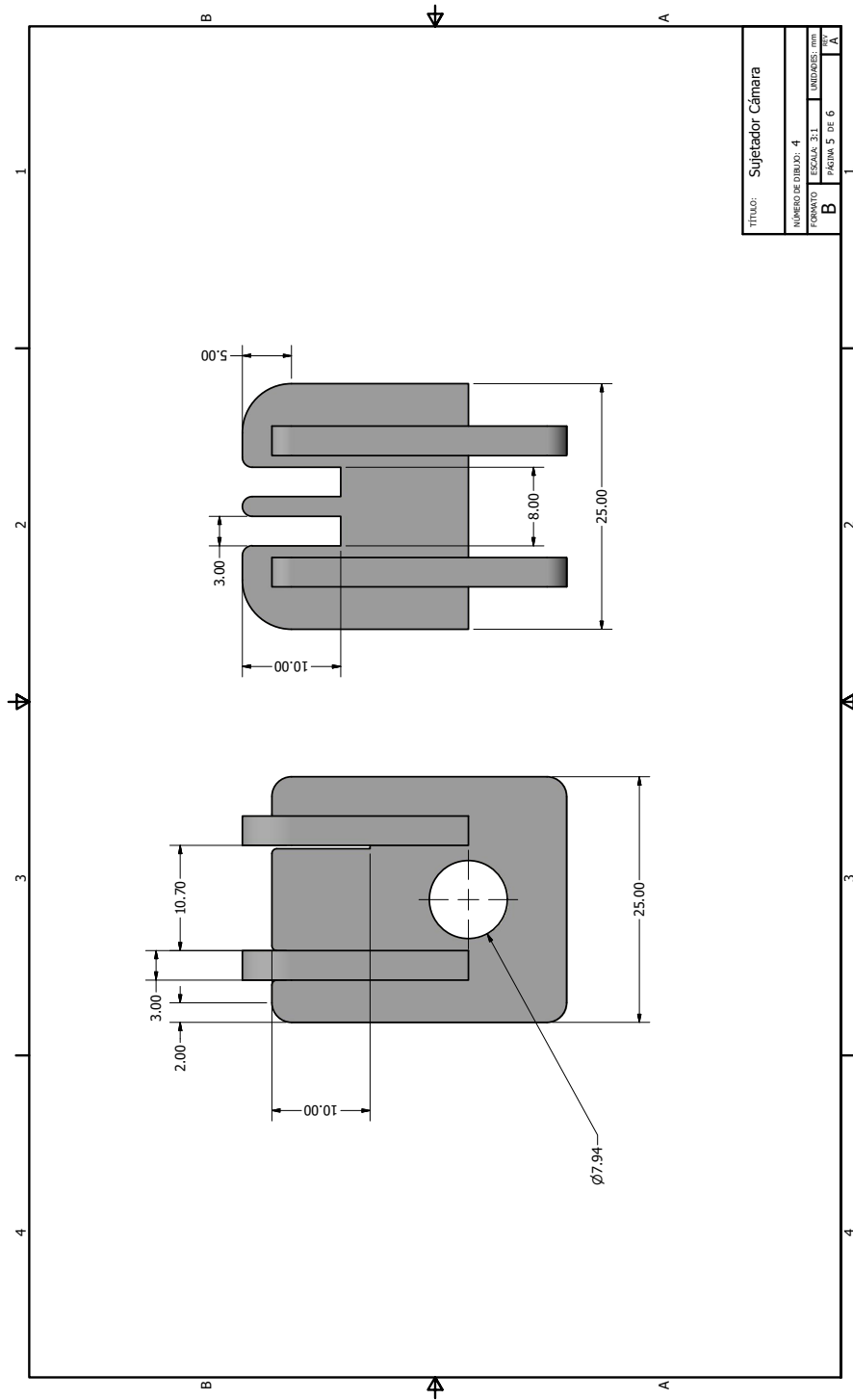


Figura 57: Plano del sistema completo Hoja 5

16.4.6. ANEXO D.6

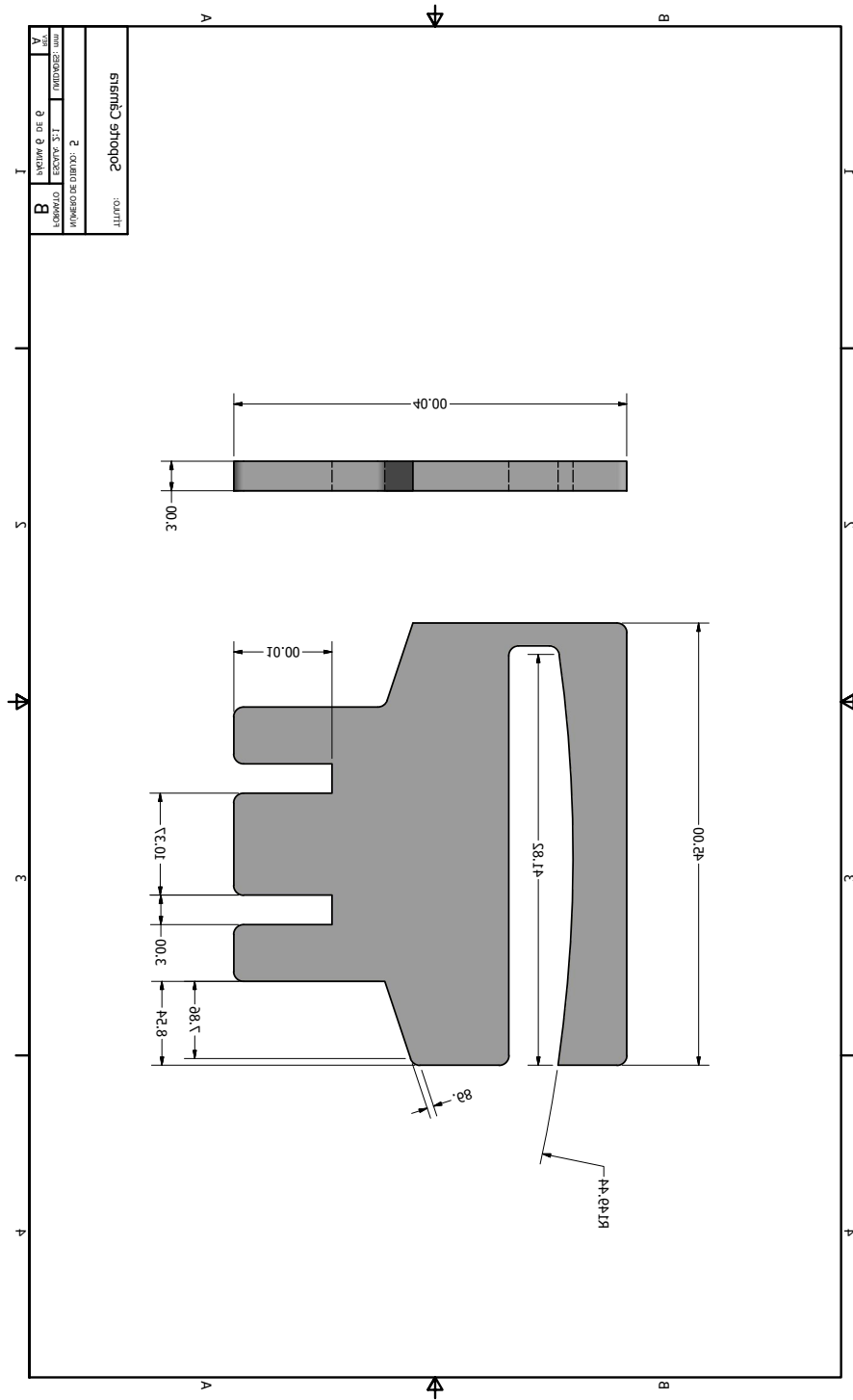


Figura 58: Plano del sistema completo Hoja 6