

# UNIVERSITAT POLITÈCNICA DE VALÈNCIA

## Escuela Técnica Superior de Ingeniería del Diseño

---

### SISTEMA SELECCIONADOR DE BOTELLAS MEDIANTE VISIÓN ARTIFICIAL EN RASPBERRY PI

### TRABAJO FINAL DEL MÁSTER UNIVERSITARIO EN INGENIERÍA MECATRÓNICA

**REALIZADO POR**

**Jhulius Castillo Suárez**

**TUTORIZADO POR**

**Carlos Ricolfe Viala**

**CURSO ACADÉMICO: 2019/2020**

**Valencia, España 30/11/2020**

## Resumen

Este trabajo fin de máster, Sistema seleccionador de botellas mediante visión artificial en Raspberry pi, realizado por D. Jhulius Castillo Suárez bajo la tutela y supervisión del Prof. D. Carlos Ricolfe Viala en Valencia, noviembre de 2020. Desarrollo y explica una aplicación de la visión artificial en un ámbito industrial.

Basándose en la utilización de equipos de bajo coste y lenguajes de programación “open source” para lograr el control de una cámara industrial y el desarrollo de una interfaz de usuario que permite configurar el sistema de una forma fácil y sencilla.

El presente proyecto pretende el diseño y elaboración de un sistema seleccionador de botellas mediante la implementación de visión artificial, utilizando una Raspberry pi 4B como controlador de todos los periféricos requeridos.

La utilidad del proyecto se centra en el desarrollo de un software capaz de detectar el nivel de líquido en botellas de vidrio transparente y la presencia del tapón de estas. también detalla los materiales requeridos para la fabricación de un prototipo que funcione controlado por una Raspberry pi.

Este proyecto busca el desarrollo de un prototipo que permita replicar la funcionalidad de máquinas de control de calidad industriales y así mostrar las posibilidades que se tienen con softwares “open source” y microprocesadores de bajo coste en el mundo de la visión artificial, más sin embargo no busca la sustitución de los controladores actuales por el utilizado en este.

## Summary

This master's thesis, Bottle sorting system using artificial vision in Raspberry pi, carried out by Mr. Jhulius Castillo Suárez under the tutelage and supervision of Prof. Mr. Carlos Ricolfe Viala in Valencia, November 2020. Development and explanation of an application of artificial vision in an industrial setting.

Based on the use of low-cost equipment and "open source" programming languages to control an industrial camera and the development of a user interface that allows the system to be configured in an easy and simple way.

This project aims to design and develop a bottle sorting system through the implementation of artificial vision, using a Raspberry pi 4B as a controller for all required peripherals.

The usefulness of the project focuses on the development of a software capable of detecting the level of liquid in transparent glass bottles and the presence of the stopper of these. It also details the materials required to make a working prototype controlled by a Raspberry pi.

This project seeks the development of a prototype that allows replicating the functionality of industrial quality control machines and thus show the possibilities that exist with "open source" softwares and low-cost microprocessors in the world of artificial vision, more however it does not seek to replace the current drivers with the one used in this one.

## Agradecimientos

Antes que nada, agradezco a Dios por permitirme dar con las personas e informaciones indicadas para poder llevar a cabo mis estudios y este proyecto.

Agradezco a mis padres, María Iluminada Suárez y Apolinar Castillo, por todo lo que han hecho y dejado de hacer para que pudiera llegar hasta aquí. Gracias por siempre apoyarme y ayudarme a seguir avanzando. También agradezco a mis hermanas, Jhuliana Castillo y Rosa Castillo, que siempre me han brindado su apoyo y cariño.

Agradezco a todos aquellos que me apoyaron consciente e inconscientemente, a aquellos que me aceptaron y me hicieron sentir como en casa, aquí, en una tierra tan lejana. Gracias a esos buenos amigos incondicionales que siempre estuvieron ahí para darme una mano: Onur Cimen, Miguel Mahiques, Guillermo Suárez, José Salvador Bellver, Daniel Naturil y Héctor Baviera (sí, tú también), sin ustedes no hubiera terminado.

Gracias a todos.

## Índice

Resumen .....	II
Summary.....	III
Agradecimientos.....	IV
<b>Índice.....</b>	<b>I</b>
<b>Índice de tablas.....</b>	<b>IV</b>
<b>Índice de ilustraciones .....</b>	<b>V</b>
<b>Abreviaturas .....</b>	<b>VII</b>
<b>1 Introducción.....</b>	<b>1</b>
<b>2 Marco teórico .....</b>	<b>2</b>
2.1 Visión artificial .....	2
2.1.1 Técnicas de procesamiento de imágenes.....	2
2.1.2 Background Subtraction .....	2
2.1.3 Filtro Thresholding.....	3
2.1.3.1 <i>Threshold Binary</i> .....	4
2.1.4 Encontrar contornos .....	4
2.2 Lenguajes de programación y librerías .....	5
2.2.1 Lenguaje de programación Python.....	5
2.2.2 Librería OpenCV.....	5
2.2.3 Librería tkinter .....	5
2.3 Raspberry Pi.....	5
2.4 Cámaras y sensores .....	6
2.4.1 Cámara industrial .....	6
2.4.2 Sensor infrarrojo.....	6
<b>3 Desarrollo y elaboración del proyecto.....</b>	<b>6</b>
3.1 Aplicación del proyecto.....	6
3.2 Descripción del proyecto.....	7
3.2.1 Modo ajuste o configuración .....	7
3.2.2 Modo producción .....	8
3.3 Elaboración y diseño.....	8
3.3.1 Pliego de condiciones.....	8
3.3.1.1 <i>Descripción del proceso de elaboración</i> .....	8
3.3.1.2 <i>Condiciones de ejecución del proyecto</i> .....	9
3.3.1.2.1 <i>Selección de dispositivos necesarios</i> .....	9
3.3.1.2.2 <i>Diseño de la estructura física</i> .....	9

3.3.1.2.3	<i>Configuración inicial de la Raspberry pi</i> .....	9
3.3.1.2.4	<i>Desarrollo de códigos</i> .....	9
3.3.1.2.5	<i>Ensamblaje y configuración del prototipo</i> .....	10
3.3.2	Equipos y/o dispositivos utilizados .....	10
3.3.2.1	<i>Raspberry</i> .....	10
3.3.2.2	<i>Cámara y objetivo</i> .....	10
3.3.2.2.1	<i>Cámara</i> .....	10
3.3.2.2.2	<i>Objetivo</i> .....	11
3.3.2.3	<i>Interruptor de sensor infrarrojo</i> .....	12
3.3.2.4	<i>Iluminación</i> .....	12
3.3.2.5	<i>Adaptador POE</i> .....	14
3.3.2.6	<i>Prototipo</i> .....	14
3.3.3	Ambiente virtual .....	16
3.3.3.1	<i>Interfaz Gráfica de Usuario (GUI)</i> .....	16
3.3.3.1.1	<i>Control de acceso</i> .....	16
3.3.3.1.2	<i>Estadísticas</i> .....	17
3.3.3.1.3	<i>Registrar</i> .....	18
3.3.3.1.4	<i>Descartadas</i> .....	19
3.3.3.1.5	<i>Ajustar</i> .....	20
3.3.3.1.6	<i>Modo operador</i> .....	23
3.3.3.2	<i>Adquisición y Análisis de imágenes</i> .....	24
3.3.3.2.1	<i>Adquisición de imágenes</i> .....	24
3.3.3.2.1.a	<i>Configuración inicial de la cámara</i> .....	24
3.3.3.2.1.b	<i>Ajuste de iluminación y enfoque</i> .....	25
3.3.3.2.1.c	<i>Adquisición</i> .....	25
3.3.3.2.2	<i>Procesamiento y análisis de imágenes</i> .....	25
3.3.3.2.2.a	<i>Definir área de interés</i> .....	26
3.3.3.2.2.b	<i>Determinar nivel de liquido</i> .....	26
3.3.3.2.2.c	<i>Determinar la presencia y posición del tapón</i> .....	28
3.3.3.2.2.d	<i>Comparar los valores obtenidos con los admisibles</i> .....	28
3.3.4	Ambiente físico .....	28
3.3.4.1	<i>Interconexión de equipos</i> .....	28
3.3.4.2	<i>Diseño de prototipo</i> .....	30
3.3.5	Análisis de costos .....	32
3.3.5.1	<i>Costo de dispositivos</i> .....	32
3.3.5.2	<i>Costo de estructura diseñada</i> .....	33
3.3.5.3	<i>Costo total</i> .....	34

<b>4 Conclusiones.....</b>	<b>35</b>
<b>5 Bibliografía .....</b>	<b>36</b>
<b>Anexos.....</b>	<b>I</b>
Anexo 1 – Diagramas de flujo .....	I
Anexo 2 – Planos .....	II
Anexo 3 – Esquema eléctrico .....	XI
Anexo 4 – Cotizaciones y justificantes de precios.....	XII
Anexo 5 – Especificaciones de dispositivos.....	XVII

## Índice de tablas

Tabla 3.1 - Configuración inicial de la Raspberry pi.....	9
Tabla 3.2 - Calculo de posibles distancias focales .....	11
Tabla 3.3 - Especificaciones del sensor E18-D80NK.....	12
Tabla 3.4 - Especificaciones técnicas del BKL1510A .....	13
Tabla 3.5 - Pestañas accesibles por Operador .....	16
Tabla 3.6 - Pestañas accesibles por Especialista.....	16
Tabla 3.7 - Parámetros de configuración inicial de la cámara .....	25
Tabla 3.8 - Equivalencia de volumen.....	27
Tabla 3.9 - Características de material utilizado .....	31
Tabla 3.10 - Costo de dispositivos .....	33
Tabla 3.11 - Costo de estructura diseñada.....	34
Tabla 3.12 - Costo total sin ingeniería ni desarrollo .....	34
Tabla 3.13 - Costo de la ingeniería y el desarrollo .....	34



## Índice de ilustraciones

Ilustración 2.1 - Mascara resultante del Background Subtractor MOG.....	3
Ilustración 2.2 - Resultado de aplicar filtro Thresholding.....	3
Ilustración 2.3 - Explicación gráfica de filtro Threshold Binary.....	4
Ilustración 2.4 - Resultado de filtro Canny y Findcontour.....	4
Ilustración 2.5 - Raspberry pi 4B .....	5
Ilustración 2.6 - Funcionamiento de sensor infrarrojo.....	6
Ilustración 3.1 - Planta llenadora de botellas.....	6
Ilustración 3.2 - Ensamble del prototipo del proyecto.....	7
Ilustración 3.3 - Esquema de funcionamiento del modo producción .....	8
Ilustración 3.4 - Esquema de GPIO de Raspberry pi.....	10
Ilustración 3.5 - Cámara industrial.....	10
Ilustración 3.6 - Longitud del cuello de la botella .....	11
Ilustración 3.7 - conexión e interruptor de sensor infrarrojo.....	12
Ilustración 3.8 - Tipo de iluminación utilizado.....	13
Ilustración 3.9 - Adaptador PoE.....	14
Ilustración 3.10 - Vistas laterales del prototipo diseñado .....	14
Ilustración 3.11 - Ventana de control de acceso .....	17
Ilustración 3.12 - Pestaña de Estadísticas .....	17
Ilustración 3.13 - Ventana para exportar datos .....	18
Ilustración 3.14 - Pestaña Registrar.....	18
Ilustración 3.15 - Pestaña Descartadas A.....	19
Ilustración 3.16 - Pestañas Descartadas B.....	20
Ilustración 3.17 - Pestaña Ajustar, botella OK.....	22
Ilustración 3.18 - Pestaña Ajustar, botella NOK.....	22
Ilustración 3.19 - Pestaña Ajustar, máscara definitiva.....	23
Ilustración 3.20 - Ventana del modo Operador .....	23
Ilustración 3.21 - Configuración inicial, pylon IP Configurator .....	24
Ilustración 3.22 - Enfoque de imagen, pylon viewer.....	25
Ilustración 3.23 - Esquema de proceso de análisis de imagen.....	26
Ilustración 3.24 - Medidas de volumen para mapeo .....	27
Ilustración 3.25 - Obtención de equivalente de volumen .....	27
Ilustración 3.26 - Esquema de conexión Ethernet y PoE.....	28
Ilustración 3.27 - Esquema de conexión eléctrica.....	29
Ilustración 3.28 - Conexión en protoboard.....	29
Ilustración 3.29 - Prototipo diseñado.....	30
Sistema seleccionador de botellas mediante visión artificial en Raspberry pi	V
TFM del Máster en Ingeniería Mecatrónica - Jhulius Castillo Suárez	

Ilustración 3.30 - Modulo de SolidWorks - Sheet Metal.....	30
Ilustración 3.31 - parámetros de doblado de la chapa.....	31
Ilustración 3.32 - Doblado de chapa .....	32
Ilustración 3.33 - Modulo SolidWorks, Structure System.....	32

## Abreviaturas

A continuación, se muestra un listado de las abreviaturas que se utilizan en este documento.

<b>BS</b>	Background Subtraction
<b>Fps</b>	Fotos por segundo
<b>GPIO</b>	General Purpose Input/Output
<b>GUI</b>	Graphical User Interface
<b>IVA</b>	Impuesto sobre el Valor Añadido
<b>LED</b>	Light-Emitting Diode
<b>LPDDR4</b>	Low Power Double Data Rate Random access memory
<b>PIR</b>	Passive Infrared Sensor
<b>PoE</b>	Power over Ethernet
<b>RAM</b>	Random Access Memory

## 1 Introducción

La visión artificial tiene una gran importancia en el sector industrial, ya que permite optimizar el control de calidad mediante inspecciones automatizadas, haciendo los procesos más rápidos y fiables, a la vez que eliminan el error humano.

La tecnología de visión beneficia a la industria manufacturera con técnicas de procesamiento de imágenes, aprendizaje automático y profundo y transferencia de datos, la utilización de cámaras de distintas tecnologías se integra cada vez más a la industria de hoy en día y todas las técnicas de procesamiento y análisis de imágenes van permitiendo el desarrollo de nuevas aplicaciones día tras día.

En la línea de fabricación o envasado es muy común la utilización de las inspecciones de visión en los productos. Donde los sistemas se utilizan desde el uso de cámaras inteligentes autónomas para llevar a cabo una inspección con la entrega de los resultados hasta los sistemas basados en PC que pueden incluir múltiples cámaras o múltiples estaciones de inspección. Los sistemas de visión son adaptables para las líneas existentes o para nuevas líneas a diseñar. También mediante métodos de control de procesos estadísticos se puede utilizar las inspecciones de visión para verificar las mediciones críticas y analizar las tendencias de estas mediciones y determinar la tolerancia.

La disponibilidad de pequeñas placas de procesamiento integradas, generalmente basadas en la arquitectura ARM, ofrece un gran potencial para el desarrollo de sistemas de visión integrados en procesos de fabricación y otros equipos. Muchas de las bibliotecas y conjuntos de herramientas de procesamiento de imágenes líderes ahora se pueden migrar a estas plataformas, ofreciendo una amplia gama de soluciones de visión en este formato. La combinación de estas capacidades de procesamiento con cámaras de bajo costo, incluidas las cámaras de nivel de placa, significa que los sistemas de visión podrían incorporarse en una amplia variedad de productos y procesos con gastos generales comparativamente pequeños.

## 2 Marco teórico

### 2.1 Visión artificial

La visión artificial es una disciplina que permite tratar las imágenes mediante computadores con procesos de adquisición, análisis y procesamiento para obtenerla información numérica o simbólica de las imágenes.

A través de los sistemas de visión artificial, es posible mejorar la producción en términos de velocidad y calidad del producto. Porque un sistema automatizado con un sistema de visión puede garantizar que los resultados tengan una mayor precisión analítica y un rango de error más bajo en tareas repetitivas. Los sistemas automatizados pueden tomar decisiones de descartar o dejar pasar un producto en base a la información dada por el sistema de visión artificial.

Finalmente se puede deducir que por medio de estos sistemas de visión se logra aumentar la calidad de producción donde, a su vez, incrementará el número de ventas de los productos y la fiabilidad del producto.

En los sistemas de visión artificial en la industria podemos distinguir varios tipos:

- Sensores de visión, se utilizan donde los sistemas de visión buscan determinar la condición de un producto. Estos son de fácil integración y bajo coste.
- Cámaras inteligentes y sistemas de visión integrados, donde destacan su fiabilidad para realizar cálculos para detectar cualquier inspección de visión con una gran capacidad de procesamiento y resolución. Estos suelen ser de fácil instalación debido a sus entradas y salidas que proporcionan una gran disponibilidad.
- Sistemas de visión avanzados, son sistemas entrenados para realizar análisis complejos en líneas de producción donde es difícil controlar todos los parámetros requeridos para un análisis.

La clave de todas las inspecciones visuales es el procesamiento de imágenes, que se pueden modificar y mejorar mediante técnicas de procesamiento para obtener la información requerida. A continuación, puede distinguir entre diferentes tecnologías de procesamiento de imágenes.

#### 2.1.1 Técnicas de procesamiento de imágenes

##### 2.1.2 Background Subtraction

EL background subtraction se basa en la detección de cambios en secuencias de imágenes y de esta forma identificar sustraer el elemento nuevo de la imagen.

Para aplicar esta técnica con OpenCV se debe crear un objeto con la función `cv2.createBackgroundSubtractorMOG()` y pasarle la imagen de fondo y la secuencia de imágenes donde aparece el objeto y esta devuelve la máscara.

La sustracción de fondo o Background Subtraction (BS), es un método utilizado para generar una máscara de primer plano y busca crear una imagen binaria que contenga los píxeles de los objetos en movimiento de la secuencia de imágenes o la imagen bajo estudio.

El BS calcula la máscara de primer plano realizando una resta entre la imagen actual y una imagen de fondo, que contiene la parte estática de la secuencia.

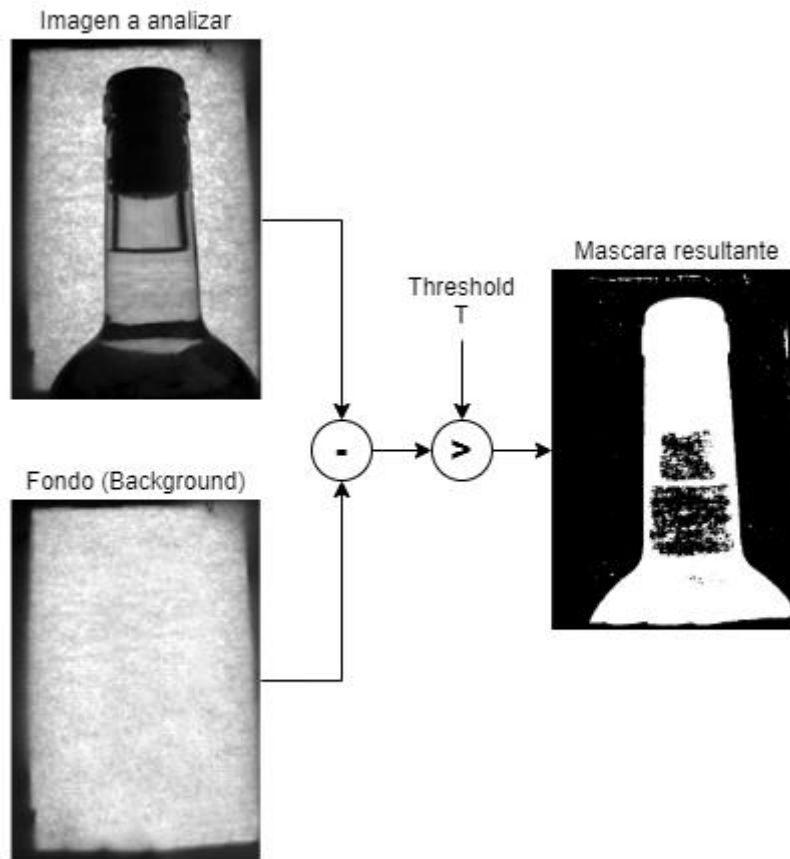


Ilustración 2.1 - Mascara resultante del Background Subtractor MOG

### 2.1.3 Filtro Thresholding

Este método se separa las partes de una imagen basado en la variación de intensidad de los píxeles del objeto bajo estudio y los píxeles de fondo.

Para diferenciar los píxeles de interés del resto, se realiza una comparación de los valores de intensidad de cada píxel con respecto a un límite establecido.

Una vez el filtro ha hecho la separación basado en la intensidad de los píxeles, se le puede asignar un valor a los píxeles rechazados y de esta forma generar la máscara que distinga los píxeles de interés.



Ilustración 2.2 - Resultado de aplicar filtro Thresholding

Con OpenCV se pueden aplicar 5 diferentes tipos de Thresholding dependiendo lo que se requiera. En escrito solo mencionaremos los utilizados en la aplicación.

### 2.1.3.1 Threshold Binary

Esta operación se puede expresar como:

$$dst(x, y) = \begin{cases} Valmax & \text{Si } src(x, y) > \text{límite} \\ 0 & \text{Cualquier otro caso} \end{cases}$$

Donde:

- $dst(x, y)$  es el píxel para analizar
- $src(x, y)$  es la intensidad del píxel.

Si la intensidad del píxel es menor al límite se hace cero y si es mayor aumenta al valor establecido.



Ilustración 2.3 - Explicación gráfica de filtro Threshold Binary

### 2.1.4 Encontrar contornos

Los contornos se pueden explicar simplemente como una curva que une todos los puntos continuos (a lo largo del límite), que tienen el mismo color o intensidad. Los contornos son una herramienta útil para el análisis de formas y la detección y reconocimiento de objetos.

Para mayor precisión, se deben utilizar imágenes binarias. Por lo que antes de aplicar las funciones correspondientes para encontrar los contornos, se debe aplicar filtros como los antes mencionados.

La función para encontrar los contornos devuelve un array con las coordenadas de todos los contornos encontrados. En la imagen se muestran todos los contornos encontrados después de aplicar un filtro "canny" y dibujar con los contornos en rojo.

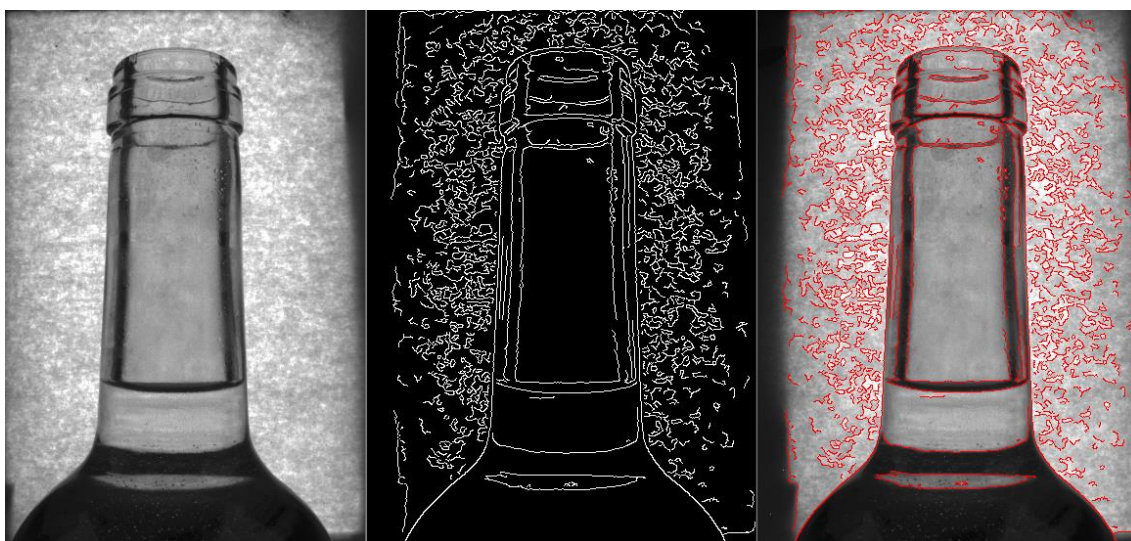


Ilustración 2.4 - Resultado de filtro Canny y Findcontour

La función utilizada para encontrar los contornos se basa en algoritmo *Topological structural analysis of digitized binary images by border following*. (Suzuki, 1985)

## 2.2 Lenguajes de programación y librerías

### 2.2.1 Lenguaje de programación Python

Python es un lenguaje de alto nivel, muy atractivo para el desarrollo de aplicaciones, además es utilizado como enlace entre aplicaciones existentes. Python admite módulos y paquetes, lo que lo hace modular y ayuda a la reutilización de código. Es un lenguaje que se puede distribuir libremente, y su biblioteca se puede adquirir en formato binario o formato fuente.

Python es popular porque permite programar de forma sencilla y rápida, permitiendo al programador que se centre en los aspectos funcionales de la aplicación.

### 2.2.2 Librería OpenCV

OpenCV (Open Source Computer Vision Library) es una biblioteca de software de visión artificial y aprendizaje automático de código abierto (Wikipedia.org, 2020). OpenCV fue creada para brindar una plataforma común para las aplicaciones de visión artificial.

Este es una librería de licencia BSD (Berkeley Software Distribution) que permite que los usuarios modifiquen el código de las funciones y bibliotecas que posee.

La biblioteca tiene más de 2500 algoritmos optimizados, que incluyen un conjunto completo de algoritmos de aprendizaje automático y visión por computadora clásicos y de última generación (OpenCV team, 2020). Los algoritmos que incorpora pueden utilizarse para la detección de objetos en imágenes o videos, a partir de funciones matemáticas que analizan los datos de la imagen dependiendo de lo que se necesite.

OpenCV tiene interfaces C ++, Python, Java y MATLAB y es compatible con Windows, Linux, Android y Mac OS (OpenCV team, 2020).

### 2.2.3 Librería tkinter

Tkinter es una librería nativa de Python que permite el diseño de interfaces de usuarios graficas. Esta se basa en posicionamiento de iconos por coordenadas y por la asignación de funciones a estos iconos.

Es una herramienta muy utilizada por desarrolladores de aplicaciones en Python.

## 2.3 Raspberry Pi

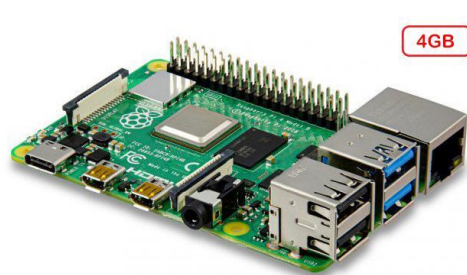


Ilustración 2.5 - Raspberry pi 4B  
(Open Circuit, 2020)

La Raspberry Pi se puede definir como un miniordenador el cual funciona bajo la plataforma Linux. Este viene equipado con puertos USB, Ethernet, micro HDMI entre otros.

Las Raspberry tienen un conjunto de entradas y salidas, llamado GPIO. Esto esta formado por 40 pines que pueden ser configurados como entradas o salidas digitales y pueden ser implementados para controlar periféricos.

La Raspberry pi 4B tiene 4gb de memoria LPDDR4 y un slot para una microSD de hasta 32gb.

Este miniordenador puede ser utilizado para muchas aplicaciones que conlleven el control de dispositivos en base a la lectura de sensores.



## 2.4 Cámaras y sensores

Las cámaras y sensores son dispositivos de entrada para un equipo de control.

### 2.4.1 Cámara industrial

Las cámaras se pueden definir como dispositivos que sirven para registrar imágenes estáticas o en movimiento. Las cámaras industriales son cámaras especiales que vienen preparadas para soportar el ambiente hostil de la industria.

Hay diferentes tipos de cámaras, con características y modos de operación diversas. Las características principales que diferencian a las cámaras industriales de las convencionales es el tipo de protección IP y la aplicación para la cual está concebida.

### 2.4.2 Sensor infrarrojo

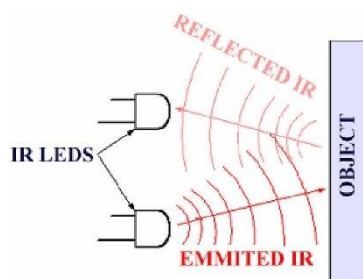


Ilustración 2.6 - Funcionamiento de sensor infrarrojo (elandroidelibre, 2020)

Un sensor de infrarrojos (IR) es un dispositivo electrónico que mide y detecta la radiación infrarroja en su entorno circundante.

Hay dos tipos de estos sensores, los activos que generan radiación y los pasivos que solo captan la radiación circundante.

Estos pueden ser utilizados como interruptores de distancia de accionamiento variables, mediante la implementación de electrónica que permita configurarlos.

Los sensores infrarrojos activos constan de dos LED infrarrojos, uno emite y el otro recibe la radiación convirtiéndola en una señal eléctrica.

## 3 Desarrollo y elaboración del proyecto

En los puntos siguientes se describe la aplicación, funcionamiento y elaboración del proyecto.

### 3.1 Aplicación del proyecto

Aunque este es un proyecto educativo y de investigación que solo busca el desarrollo de un prototipo que permita replicar la funcionalidad de máquinas de control de calidad industriales y así mostrar las posibilidades que se tienen con softwares "open source" y microprocesadores de bajo coste en el mundo de la visión artificial. Se ha buscado replicar una aplicación real de la industria.

El campo de aplicación del proyecto se encuentra en las líneas de llenado o embotelladoras, con botellas de cristal translúcido.

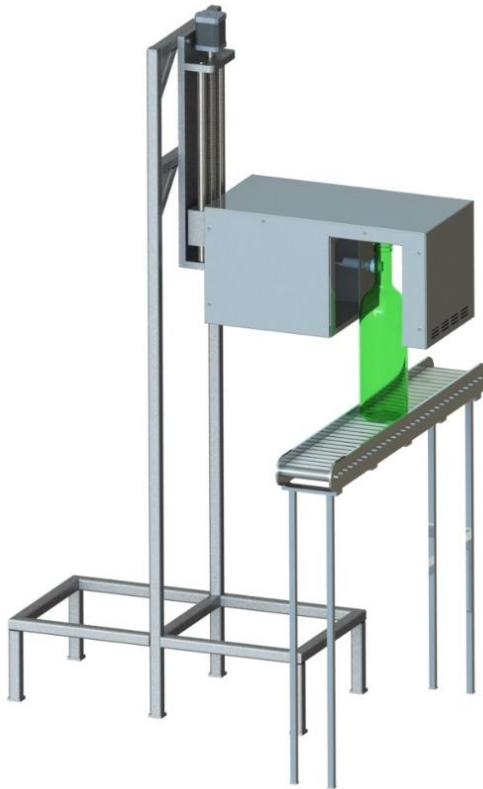
La configuración que se ha dado ha sido para la detección del nivel de líquido y la presencia de corcho de las botellas de vino blanco o cullas botellas cumplan con lo antes mencionado.



Ilustración 3.1 - Planta llenadora de botellas (Boca, 2020)

El sistema es un control de calidad para supervisar que las botellas tengan el volumen de líquido correcto y que la tapa o tapón haya si bien colocado. Mas sin embargo no se pretende que el prototipo presentado sustituya las máquinas de control de calidad existentes, puesto que harían falta equipos periféricos al microcontrolador utilizado para aumentar su fiabilidad y confiabilidad en un ambiente industrial y dichos periféricos no se están tomando en cuenta en este trabajo.

### 3.2 Descripción del proyecto



*Ilustración 3.2 - Ensamble del prototipo del proyecto*

El proyecto en cuestión es el diseño de una seleccionadora de botellas controlada por una Raspberry Pi 4B. Este se divide en tres partes principales:

- Selección de equipos
- Diseño y programación de la interfaz de usuario.
- Programación de las funciones de análisis de imágenes.
- Diseño mecánico e interconexión

El proceso de selección se basa en el análisis de las imágenes tomadas por una cámara industrial controlada por la Raspberry.

Se tienen dos modos de operación, el modo ajuste y el modo producción.

#### 3.2.1 Modo ajuste o configuración

Este modo es el que permite al usuario, con los permisos adecuados, configurar los parámetros de adquisición de la cámara:

- Tiempo de exposición
- Ganancia
- Nivel de negros
- Cambio digital (shift digital)
- Compensación (offset) en X e Y

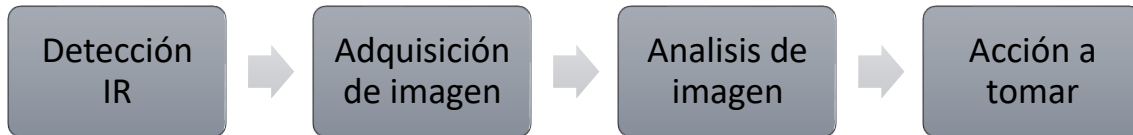
también permite establecer los parámetros de nivel máximo y mínimo con los cuales se compara el nivel para determinar si la botella es descartada o no.

Una vez configurado el sistema, el usuario puede guardar la configuración para que se cargue a la cámara cada vez que el sistema sea iniciado.

Este modo pone la cámara en modo “video” y permite ver un análisis en tiempo real de la botella.

### 3.2.2 Modo producción

En el modo producción el sistema procede a analizar las botellas que van pasando, por la línea transportadora, basado en los parámetros configurados y en los cargados previamente referentes a las dimensiones del tapón y la botella.



*Ilustración 3.3 - Esquema de funcionamiento del modo producción*

El proceso inicia con la detección de la botella por parte del sensor infrarrojo (IR), quien activa la adquisición de imagen.

Una vez se tiene la imagen, la misma es analizada por funciones de procesamiento de imágenes de OpenCV y Python, y los datos obtenidos por estas funciones son comparados con los parámetros previamente configurados y de esta forma establecer si la botella merece ser descartada o no.

El sistema está configurado para descartar la botella si se presenta uno o más de los siguientes casos:

- Volumen fuera de rango
- Corcho mal colocado
- Ausencia de corcho

Cada vez que una botella es analizada el resultado del análisis es mostrado en una tabla, indicando si fue o no descartada, y en caso de que se haya descartado, el motivo por el cual se ha descartado. Las informaciones de esta tabla pueden ser exportadas en forma “.csv”.

## 3.3 Elaboración y diseño

A continuación, se describen todas las partes y/o etapas que conformaron la elaboración de este proyecto.

### 3.3.1 Pliego de condiciones

El presente proyecto pretende el diseño y elaboración de un sistema seleccionador de botellas mediante la implementación de visión artificial, utilizando una Raspberry pi 4B como controlador de todos los periféricos requeridos.

La utilidad del proyecto se centra en el desarrollo de un software capaz de detectar el nivel de líquido en botellas de vidrio transparente y la presencia del tapón de estas. también detalla los materiales requeridos para la fabricación de un prototipo que funcione controlado por una Raspberry pi.

Se busca construir un prototipo que permita replicar la funcionalidad de máquinas de control de calidad industriales y así mostrar las posibilidades que se tienen con softwares “open source” y microprocesadores de bajo coste en el mundo de la visión artificial, más sin embargo no busca la sustitución de los controladores actuales por el utilizado en este.

#### 3.3.1.1 Descripción del proceso de elaboración

Las partes que componen el proyecto son las siguientes:

1. Selección de dispositivos necesarios.
2. Diseño de la estructura física.
3. Comprar y/o encargo de dispositivos y estructura.
4. Configuración inicial de la Raspberry.
5. Desarrollo del código de la interfaz de usuario.
6. Desarrollo del código de análisis de botellas.
7. Ensamblaje y configuración del prototipo.
8. Realización de pruebas.

Todas estas partes unidas conforman el proyecto y cada una de estas deben ser llevadas a cabo por personas capacitadas para la realización de cada parte.

### 3.3.1.2 Condiciones de ejecución del proyecto

En este apartado se indican las pautas para la elaboración del proyecto.

#### 3.3.1.2.1 Selección de dispositivos necesarios

La selección de los dispositivos es lo primero que se deberá hacer, ya que de estos dependen las dimensiones de la estructura física y más partes del proyecto. Se debe verificar que lo calculado y/o seleccionado se corresponda con equipos comerciales disponibles y en caso de que no sea el caso, se debe recalculer y hacer una nueva selección.

#### 3.3.1.2.2 Diseño de la estructura física

Una vez seleccionados los dispositivos principales: cámara, lente y luz. Se debe calcular a que distancia de enfoque la cámara capta el área útil de la luz a utilizar. A partir de esta distancia se deben definir las demás medidas de la estructura de forma que permita el correcto posicionamiento de los demás dispositivos.

#### 3.3.1.2.3 Configuración inicial de la Raspberry pi

La configuración de la Raspberry conlleva la instalación del sistema operativo y todos los softwares y librerías requeridas para el correcto funcionamiento del código a desarrollar. Los software, librerías y aplicaciones son las ...

No.	Tipo	Nombre	versión
1	Software	pylon Viewer 32bit	6.2.3.8556
2	Software	Pylon IP Configurator	-
3	Lenguaje	Python	3.7.3
4	Librería	OpenCV	4.3
5	Librería	numpy	1.16.2
6	Librería	Pillow	5.4.1
7	Librería	pypylon	1.6.0
8	Librería	imutils	0.5.3
9	Librería	xml.dom.minidom	0.0.0

Tabla 3.1 - Configuración inicial de la Raspberry pi

#### 3.3.1.2.4 Desarrollo de códigos

Para poder iniciar el desarrollo del código requerido es necesario que todas las librerías antes mencionadas estén correctamente instaladas. Se debe seguir un formato de programación estructurado con el mismo sistema de declaración de variables y constantes en todo el código.

Se deben comentar todas las funciones para que cualquier otra persona que tenga acceso al código, con el conocimiento adecuado, pueda entenderlo.

### 3.3.1.2.5 Ensamblaje y configuración del prototipo.

Esta etapa puede ser realizada en paralelo con el desarrollo del código, para poder ir haciendo pruebas a medida que se va avanzando.

## 3.3.2 Equipos y/o dispositivos utilizados

Para la elaboración y prueba de este proyecto hicieron falta la utilización de dispositivos para llevar a cabo las pruebas y confirmar que el código funciona. Los equipos y dispositivos principales se enumeran a continuación.

### 3.3.2.1 Raspberry

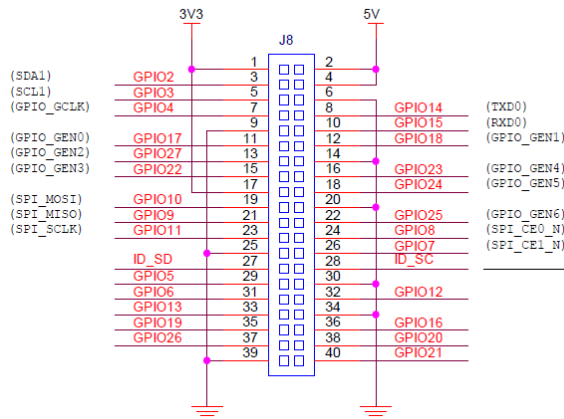


Ilustración 3.4 - Esquema de GPIO de Raspberry pi

Como ya se ha mencionado, el equipo controlador en este proyecto es una Raspberry pi 4B de 4gb de RAM.

Esta placa cuenta con un procesador Broadcom BCM2711, quad-core Cortex-A72 (ARM v8) 64-bit SoC @ 1.5GHz y 4gb de memoria LPDDR4, que dan abasto para el procesamiento de imágenes y las tareas que se requieren para la elaboración del proyecto.

Posee un Standard 40-pin GPIO header, que pueden ser configurados como entradas o salidas digitales y con las cuales se puede controlar la cámara que se utilizará.

Cuenta con conexión Gigabit Ethernet, lo que permite la comunicación y transferencia de datos con la cámara.

Ver anexo 5 para más detalles.

### 3.3.2.2 Cámara y objetivo

#### 3.3.2.2.1 Cámara



Ilustración 3.5 - Cámara industrial

En este proyecto se ha optado por controlar una cámara, que, a la fecha, se utiliza en aplicaciones de visión artificial en la industria. Este tipo de cámaras normalmente es utilizada en conjunto a un ordenador y un PLC para ejecutar el proceso de análisis y ejecución de proceso de descarte.

La cámara utilizada es una Basler acA640-100gm. Una cámara a blanco y negro con una resolución máxima de 659x494 píxeles y una velocidad de obturación de 100 fps.

Esta cámara se conecta con el controlador a través de la interfaz Gigabit Ethernet y también se alimenta a través del mismo conector, con PoE.

La acA640 se sincroniza con la Raspberry mediante la conexión Ethernet, de esta forma el disparo de la cámara se hace vía software. Otro parámetro importante para la elaboración de proyecto es el que tenga salida de imagen en formato YUV 422 Packed, que codifica la imagen como si fuera a color.

Ver anexo 5 para más detalles.

### 3.3.2.2.2 Objetivo

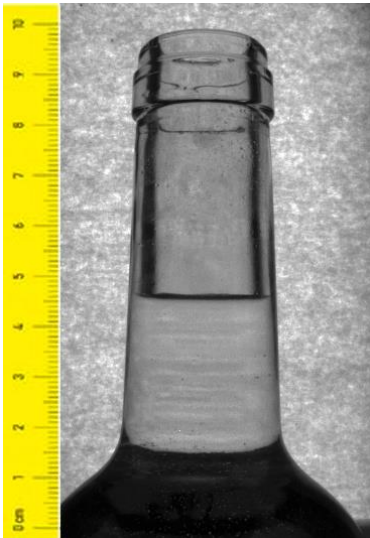


Ilustración 3.6 - Longitud del cuello de la botella

Para seleccionar el objetivo o lente para la cámara se partió de las dimensiones del objeto que se iba a estudiar, el cuello de la botella. La medida aproximada del área de estudio es de unos 110x30 mm y en base a esto se buscó el objetivo con la menor distancia de enfoque posible (10 mm).

Con la información de que el sensor de la cámara es de ¼", lo que equivale a un sensor de 3,6x2,7 mm, se procedió a calcular las posibles distancias focales del objetivo partiendo de la relación que existe entre los triángulos que se forman entre el objeto y el cristal del objetivo y entre el cristal del objetivo y el sensor. La ecuación utilizada ha sido:

$$\text{Distancia focal} = \frac{\text{Distancia de enfoque} * \text{Ancho sensor}}{\text{Altura de objeto}}$$

En nuestro caso, la cámara se ha colocado de forma horizontal por lo que el ancho del sensor se corresponde con la altura del objeto.

Los resultados de este análisis fueron:

	Ancho sensor (mm)	Distancia de enfoque (mm)	Altura Objeto (mm)	Distancia focal (mm)
1	3,6	100	110	3,27
2		110		3,60
3		120		3,93
4		130		4,25
5		140		4,58

Tabla 3.2 - Calculo de posibles distancias focales

Con estos datos, procedimos a utilizar el seleccionar de lentes de la marca de la cámara. Este mediante la introducción de las dimensiones antes mencionadas filtra todos los lentes, compatibles con la cámara, que se ajusten a las necesidades antes mencionadas. De ahí se obtuvo que para esta aplicación se debía seleccionar un lente entre las posiciones 4 y 5, y por tales motivos se eligió el MVL5WA de 4,5 mm EFL, f/1.4.

[https://lh3.googleusercontent.com/-yC\\_kU5fh-Kc/WI9mO5Kbcfl/AAAAAAAAOQM/sRLuw9g3OKM/image\\_thumb%25255B3%25255D.png?imgmax=800](https://lh3.googleusercontent.com/-yC_kU5fh-Kc/WI9mO5Kbcfl/AAAAAAAAOQM/sRLuw9g3OKM/image_thumb%25255B3%25255D.png?imgmax=800)

### 3.3.2.3 Interruptor de sensor infrarrojo

Para dar la señal de disparo se ha utilizado un sensor infrarrojo, el cual se encarga de detectar la presencia de la botella y enviar la señal de disparo a la Raspberry y esta su vez a la cámara.

El sensor utilizado ha sido el E18-D80NK, el cual es un sensor fotoeléctrico transmisor – receptor al cual se le puede ajustar la distancia de detección mediante un potenciómetro que trae incorporado.

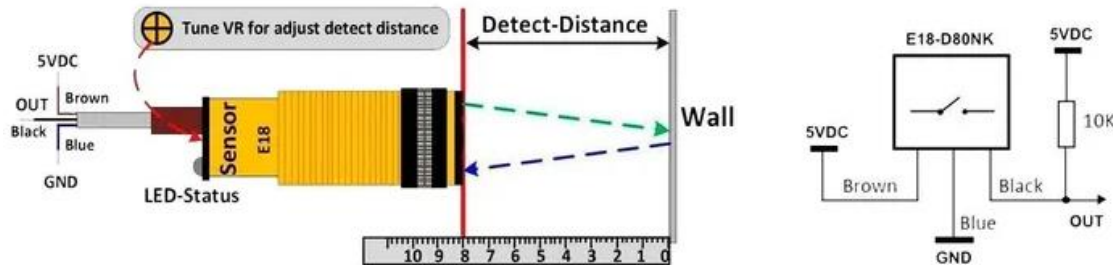


Ilustración 3.7 - conexión e interruptor de sensor infrarrojo

[https://articulo.mercadolibre.com.mx/MLM-715257695-e18-d80nk-sensor-de-proximidad-3-80cm-arduino-\\_JM](https://articulo.mercadolibre.com.mx/MLM-715257695-e18-d80nk-sensor-de-proximidad-3-80cm-arduino-_JM)

[https://images-na.ssl-images-amazon.com/images/I/61uIbUue2L.\\_AC\\_SL1001\\_.jpg](https://images-na.ssl-images-amazon.com/images/I/61uIbUue2L._AC_SL1001_.jpg)

Las características del interruptor se pueden ver en la Tabla 3.3:

Modelo	E18-D80NK
Voltaje de alimentación (VDC)	5
Corriente (mA)	100
Distancia de detección (mm)	30-800
Diámetro (mm)	18
Longitud (mm)	45
Longitud del cable (mm)	45,5
Peso (gr)	28

Tabla 3.3 - Especificaciones del sensor E18-D80NK

### 3.3.2.4 Iluminación

Las aplicaciones de visión artificial necesitan de una buena iluminación que resalten los píxeles de interés en la imagen, que para nuestro caso son el corcho y el nivel de líquido. también se debe conseguir que la fuente de luz no produzca reflejos en la superficie a analizar, para que el procesamiento de la imagen sea más sencillo.

En vista de la naturaleza del material de la botella (cristal) y lo que se está buscando detectar se ha optado por una iluminación desde atrás (backlit), en la cual el objeto queda en entre la cámara y la fuente de luz.



Ilustración 3.8 - Tipo de iluminación utilizado

Para la elaboración del prototipo y las pruebas se utilizó una luz fija hecha a partir de una bombilla LED, un sócalo y una campana difusora con las medidas específica de luz industrial. Para la implementación de este proyecto en la industria se indica el uso de la lampara industrial BKL1510A con tecnología iBlueDrive la cual permite que se pueda utilizar como estrobo, activándose solo en el momento en que se hace la adquisición de imagen.

Las características de esta luz son:

<b>Modelo</b>	<b>BKL1510A-W00i</b>
<b>Dimensiones (mm)</b>	116x154x22
<b>Superficie activa (mm)</b>	100x150
<b>Peso (g)</b>	345
<b>Grado de protección IP</b>	IP40
<b>Agujeros de montura</b>	(x7)M4I6
<b>Conexión</b>	2P male chassis connector PIN 1 = +24V PIN 2 = 0V
<b>Cable de alimentación</b>	VCB Series
<b>Tecnología iBlueDrive</b>	inline
<b>iBlueDrive connection</b>	3P aerial male inline connector. L=715mm. PIN 1 = +24V PIN 2 = 0V PIN 3 = Control
<b>Consumo continuo (W)</b>	6,8
<b>Consumo iBlueDrive(W)</b>	17 [48/11]

Tabla 3.4 - Especificaciones técnicas del BKL1510A



### 3.3.2.5 Adaptador POE



Para alimentar la cámara se ha hecho uso de un adaptador PoE, de esta forma la cámara ha podido ser controlada por el mismo cable que la alimenta el Ethernet.

Este soporta el Gigabit Ethernet, por lo que su uso no limita la transferencia de datos entre la cámara y el Raspberry.

[https://static.tp-link.com/TL-POE150S\\_UN\\_4.0\\_01\\_1501551840169h.jpg](https://static.tp-link.com/TL-POE150S_UN_4.0_01_1501551840169h.jpg)

Ilustración 3.9 - Adaptador PoE

### 3.3.2.6 Prototipo

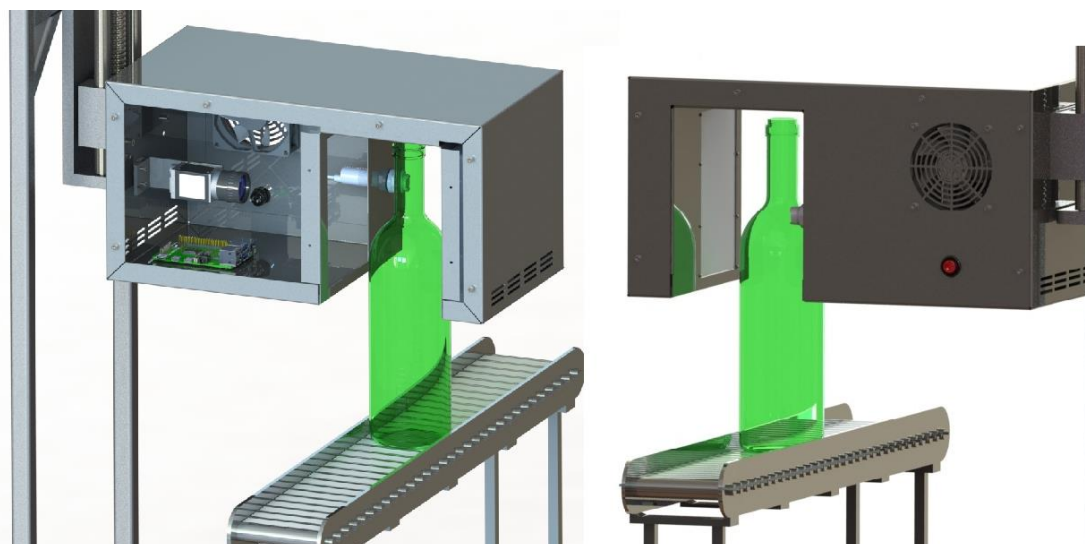

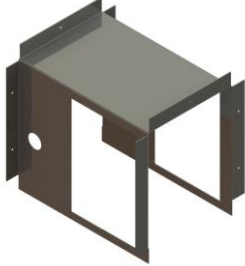

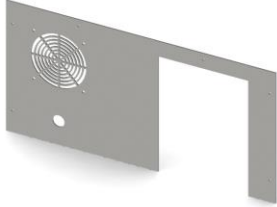




Ilustración 3.10 - Vistas laterales del prototipo diseñado

Para la integración de todos los dispositivos se diseñó (3.3.4.2 Diseño del prototipo) un prototipo basado en las distancias mínimas de enfoque y cobertura del objetivo, las dimensiones de las botellas y la luz recomendada.

Nombre	Imagen	Descripción
Carcasa exterior		<p>Esta va sostenida a la base móvil, y en la parte inferior soporta la Raspberry y demás circuitos.</p> <p>En la parte trasera soporta la cámara.</p>

<p><b>Carcasa interior</b></p>		<p>Parte interior de la carcasa, soporta la pantalla de iluminación, el interruptor infrarrojo y una mica transparente para proteger los equipos en lado izquierdo de la carcasa.</p>
<p><b>Lateral izquierdo</b></p>		<p>Tapa de cierre del lado izquierdo del equipo, esta parte es la que se asume como punto de acceso para poder manipular los que equipos en el interior de la caja.</p>
<p><b>Lateral derecho</b></p>		<p>Tapa de cierre del lado derecho, esta no debe quitarse ya que lleva un ventilador y un botón de encendido.<sup>1</sup></p>
<p><b>Base</b></p>		<p>Base que soporta la caja el tornillo sinfín que permite el ajuste vertical de esta.<sup>2</sup></p>
<p><b>Tornillo sin fin</b></p>		<p>Tornillo sin fin que permite el ajuste vertical de la caja.<sup>3</sup></p>

<sup>1</sup> El ventilador y el pulsador no se instalaron en la versión física.

<sup>2</sup> La base no se reprodujo en la versión física.

<sup>3</sup> El tornillo sin fin no se reprodujo en versión física y el control de motor no se estudia en este proyecto.

### 3.3.3 Ambiente virtual

En este proyecto se ha desarrollado un software programado en Python 3.7 y OpenCV 4.3. Se han desarrollado funciones de visión artificial que permiten analizar áreas específicas de la botella y determinar la posición del nivel y la presencia o ausencia del tapón.

A la vez se ha desarrollado una Interfaz Gráfica de Usuario que permite configurar y visualizar los resultados de los análisis.

A continuación, se desarrollan de forma detallada los dos aspectos del ambiente virtual.

#### 3.3.3.1 Interfaz Gráfica de Usuario (GUI)

La GUI se ha programado haciendo uso de la librería nativa de Python tkinter. Esta consta de diferentes ventanas las cuales tienen distintas funciones.

En la GUI se ha implementado un control de pestañas para que cada tipo de usuario tenga una ventana con múltiples pestañas. Cada pestaña se define como una clase del tipo “tk.Frame” y el acceso de cada usuario es el siguiente:

- Operador:

Pestañas de acceso del operador	
Nombre para usuario	Nombre en código
Estadísticas	WinStatistic(tk.Frame)
Descartadas	WinErrorView(tk.Frame)

Tabla 3.5 - Pestañas accesibles por Operador

- Especialista:

Pestañas de acceso del especialista	
Nombre para usuario	Nombre en código
Estadísticas	WinStatistic(tk.Frame)
Descartadas	WinErrorView(tk.Frame)
Registrar	WinRegistry(tk.Frame)
Ajustar	WinSetting(tk.Frame)

Tabla 3.6 - Pestañas accesibles por Especialista

A continuación, se explica el funcionamiento de cada venta a la vez que se destacan los aspectos principales de la programación.

##### 3.3.3.1.1 Control de acceso

Esta es la primera ventana que surge al momento de iniciar la aplicación, en esta se distinguen dos niveles de usuarios, operador y especialista.

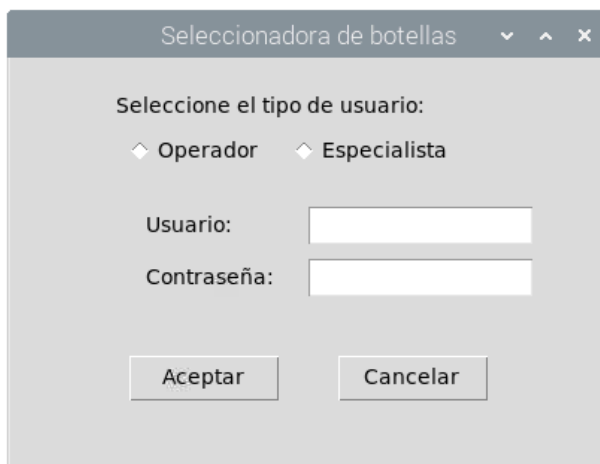
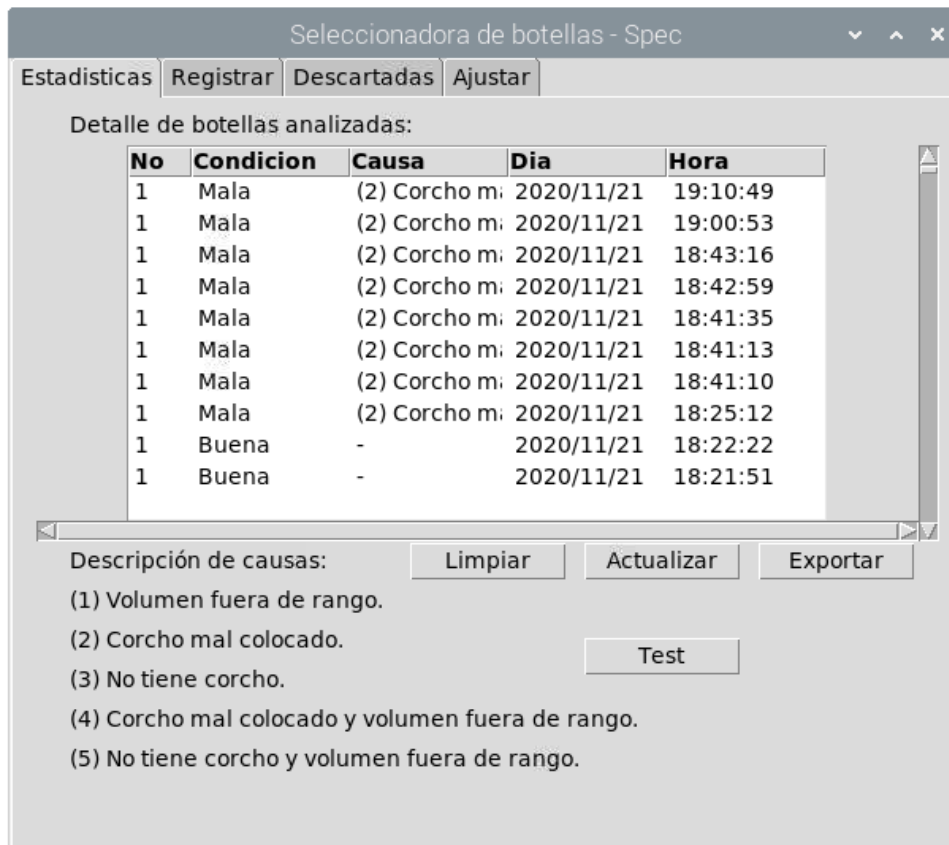


Ilustración 3.11 - Ventana de control de acceso

Al pulsar el botón “Aceptar” se ejecuta la función “LoginCheck()” la cual compara los valores introducidos con los ficheros que se encuentran en la carpeta Users. En caso de que los datos introducidos no coincidan con los contenidos en el fichero se despliega una venta de error indicando cual es la información errónea. La cual puede ser que el usuario no sea el correcto, que la contraseña no coincida o que el tipo de usuario seleccionado no sea el indicado.

### 3.3.3.1.2 Estadísticas

En la ventana estadística se muestra una lista de todas las botellas que se han analizado indicando la fecha y la hora en que la botella fue analizada, y si la botella fue descartada o no. En caso de que la botella se haya descartado, también se indica el motivo del descarte.



No	Condicion	Causa	Dia	Hora
1	Mala	(2) Corcho m:	2020/11/21	19:10:49
1	Mala	(2) Corcho m:	2020/11/21	19:00:53
1	Mala	(2) Corcho m:	2020/11/21	18:43:16
1	Mala	(2) Corcho m:	2020/11/21	18:42:59
1	Mala	(2) Corcho m:	2020/11/21	18:41:35
1	Mala	(2) Corcho m:	2020/11/21	18:41:13
1	Mala	(2) Corcho m:	2020/11/21	18:41:10
1	Mala	(2) Corcho m:	2020/11/21	18:25:12
1	Buena	-	2020/11/21	18:22:22
1	Buena	-	2020/11/21	18:21:51

Descripción de causas:

- (1) Volumen fuera de rango.
- (2) Corcho mal colocado.
- (3) No tiene corcho.
- (4) Corcho mal colocado y volumen fuera de rango.
- (5) No tiene corcho y volumen fuera de rango.

Ilustración 3.12 - Pestaña de Estadísticas

Los botones que posee tienen las siguientes funciones:

- Limpiar: Elimina todos los datos del registro, cada vez que se hace clic en el despliega una advertencia indicando que eliminará los registros y es recomendable que exporte antes de continuar.

Con el comando “open” el “get\_childre()” de los elementos ttk.Treview se ejecuta un bucle “for” para eliminar todos los elementos del archivo.

- Exportar: Este botón permite exportar el registro en formato .scv mediante el despliegue de una ventana de búsqueda en el sistema.

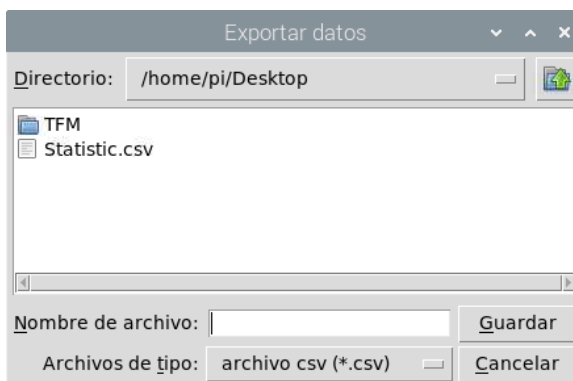


Ilustración 3.13 - Ventana para exportar datos

- Test: Este permite ejecutar un disparo de prueba de la cámara y analizar la imagen. Ejecuta las mismas funciones que el sensor infrarrojo.

### 3.3.3.1.3 Registrar

Esta ventana, a la que solo tienen acceso los especialistas, se pueden agregar nuevos usuarios.

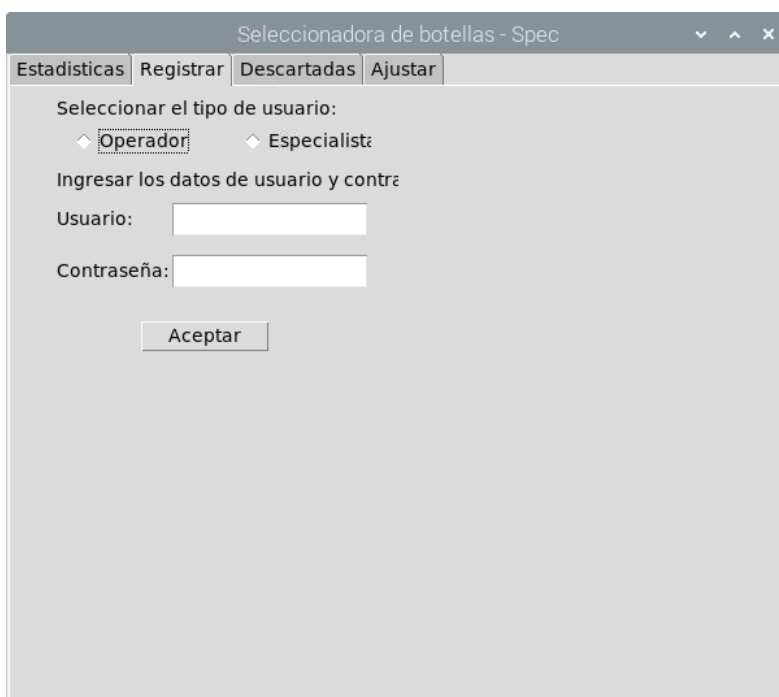


Ilustración 3.14 - Pestaña Registrar

Al hacer clic en el botón “Aceptar” se ejecuta la función “RecordUser()” la cual obtiene los valores introducidos y los escribe en un archivo independiente de la carpeta Users.

#### 3.3.3.1.4 Descartadas

En esta venta se pueden visualizar todas las botellas que han sido descartadas, con las marcas del análisis que se ha realizado. En las imágenes se pueden ver las líneas del nivel medido, los límites de nivel establecidos y el rectángulo que indica la presencia del tapón.

Las líneas se visualizarán de distintos colores dependiendo del motivo por el cual se ha descartado la botella. Los motivos y las marcas son las siguientes:

- Ausencia del tapón, en este caso no presentará ninguna marca (rectángulo) en el área correspondiente.
- Tapón mal colocado, mostrara un rectángulo de color rojo señalando el tapón.
- Nivel fuera de rango, se podrá visualizar el nivel en color rojo fuera de las líneas límites de color azul.

Todas las imágenes que se presente en esta ventana deben tener por lo menos uno de los fallos antes mencionados.

Además, también se da el nombre de la imagen, el cual indica la hora y la fecha en que fue tomada la foto y para poder cambiar de foto se cuenta con dos botones, Siguiente y Anterior, que permiten navegar por todas las fotos de la carpeta “Descarted”.

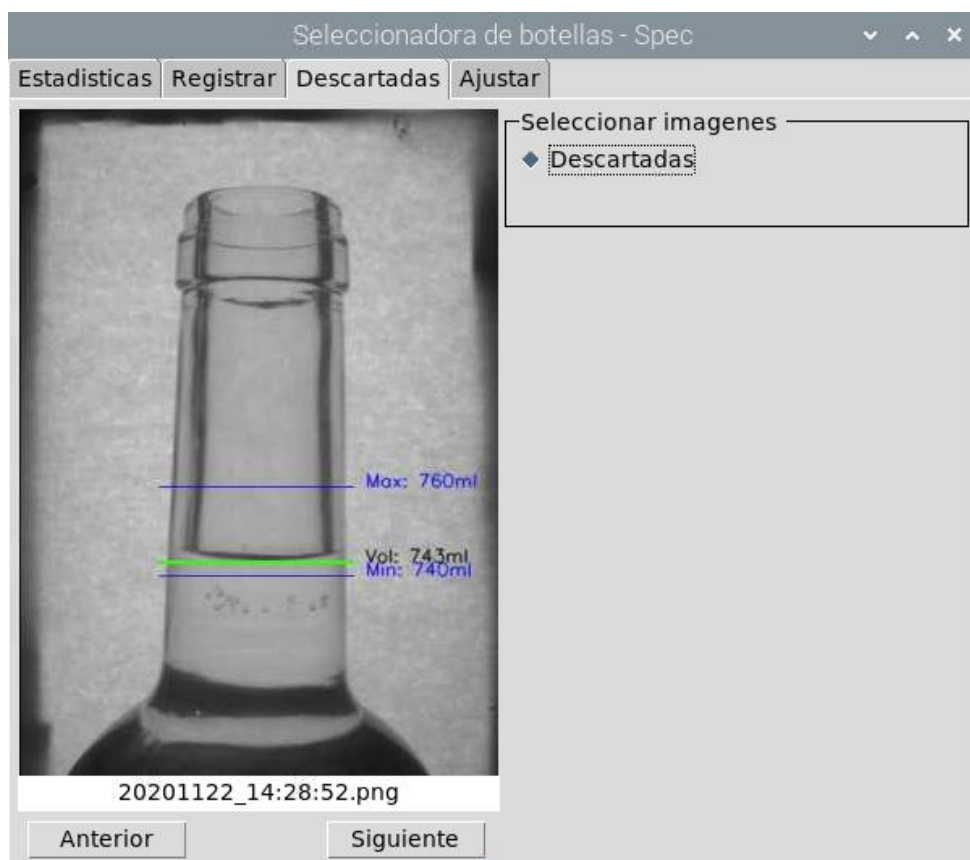


Ilustración 3.15 - Pestaña Descartadas A

Al momento de seleccionar botón radiar en la parte superior izquierda, se ejecuta la función “getImgList(Path, ext)”. Esta función genera una lista, haciendo unos de Librería “os”, con todos los elementos con tenido en “Path” que tengan la extensión establecida. En nuestro caso .png.

Al hacer clic en el botón siguiente o anterior, se ejecuta la función “getImgOpen(Window,seq)” la cual muestra en pantalla la imagen sucesiva o anterior en la lista.

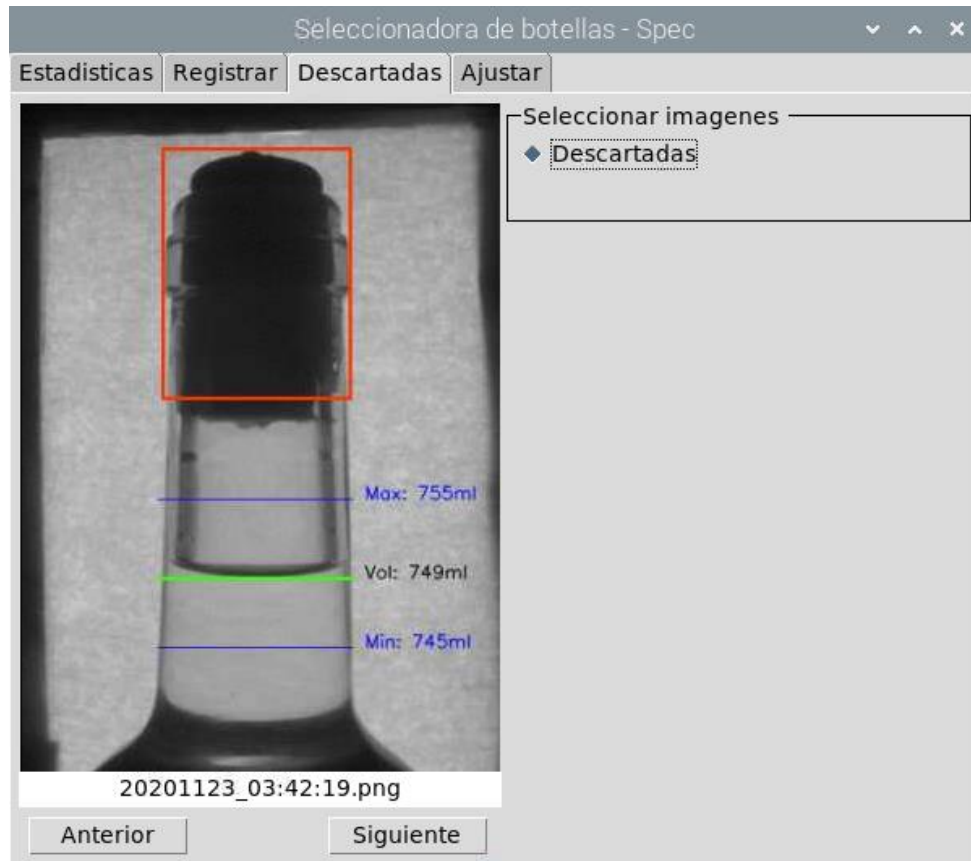


Ilustración 3.16 - Pestañas Descartadas B

### 3.3.3.1.5 Ajustar

La ventana ajustar permite establecer los parámetros de configuración de la cámara logrando que el sistema se ajuste a diferentes condiciones de iluminación, además se pueden modificar los límites de nivel mínimo y máximo. Los parámetros de la cámara que se pueden ajustar son:

- Tiempo de exposición: Este es el parámetro “ExposureTimeRaw” según el fabricante, el cual permite mantener por más o menos tiempo el paso de luz al sensor. Los comandos utilizados para configurar este parámetro son:
  - camera.ExposureTimeRaw.SetValue(10000)
  - double d = camera.ExposureTimeRaw.GetValue()
  - double d = camera.ExposureTimeRaw.GetMin()
  - double d = camera.ExposureTimeRaw.GetMax()
  
- Ganancia: Este es el parámetro “GainRaw” según el fabricante, el cual permite aumentar o disminuir la sensibilidad del sensor de imagen. Permite modificar el ISO. Los comandos utilizados para configurar es parámetro son:
  - camera.GainRaw.SetValue(10000)
  - double d = camera.GainRaw.GetValue()

- double d = camera. GainRaw.GetMin()
  - double d = camera. GainRaw .GetMax()
- Nivel de negros: Este es el parámetro “BlackLevelRaw” según el fabricante, el cual permite ajustar el nivel de negros de la cámara.
- camera. BlackLevelRaw.SetValue(10000)
  - double d = camera. BlackLevelRaw.GetValue()
  - double d = camera. BlackLevelRaw.GetMin()
  - double d = camera. BlackLevelRaw.GetMax()
- Cambio digital: Este es el parámetro “DigitalShift” según el fabricante, el cual permite multiplicar los valores de la imagen, aumentando el brillo de esta.
- camera. DigitalShift.SetValue(10000)
  - double d = camera. DigitalShift.GetValue()
  - double d = camera. DigitalShift.GetMin()
  - double d = camera. DigitalShift.GetMax()
- Compensación en X: Este es el parámetro “OffsetX” según el fabricante, el cual permite cambiar los pixeles que capta la cámara cuando se usa a una resolución menor a la nominal en la horizontal.
- camera. OffsetX.SetValue(10000)
  - double d = camera. OffsetX.GetValue()
  - double d = camera. OffsetX.GetMin()
  - double d = camera. OffsetX.GetMax()
- Compensación en Y: Este es el parámetro “OffsetY” según el fabricante, el cual permite el cual permite cambiar los pixeles que capta la cámara cuando se usa a una resolución menor a la nominal en la vertical.
- camera. OffsetY.SetValue(10000)
  - double d = camera. OffsetY.GetValue()
  - double d = camera. OffsetY.GetMin()
  - double d = camera. OffsetY.GetMax()

además de los para metros de la cámara, también permite modificar los valores del nivel mínimo y máximo que utilizan para comparar el nivel de líquido en la botella.

Debajo de la imagen hay 4 botones radiales que permiten cambiar de modo de ajuste al modo de producción y cambiar la imagen que se visualiza por la máscara que se ha aplicado.

En el modo producción no se despliega ninguna imagen o solo se muestra la última tomada en el modo Ajustar.

En el modo ajustar se inicia un video en tiempo real de lo que está frente a la cámara, de esta forma se puede ver cómo afectan la modificación de los parámetros sobre nuestra imagen y si el sistema está detectando o no lo que debe detectar.

El modo ajustar está pensado para que primero se ajuste la máscara y luego la búsqueda de nivel.

En esta ventana también hay dos botones, “Config” y “Fondo” que permiten guardar la configuración o ajuste de la cámara para que se aplique cada vez que se inicia el sistema y tomar una foto del fondo (esta foto luego se utiliza para el análisis de las imágenes con botellas).



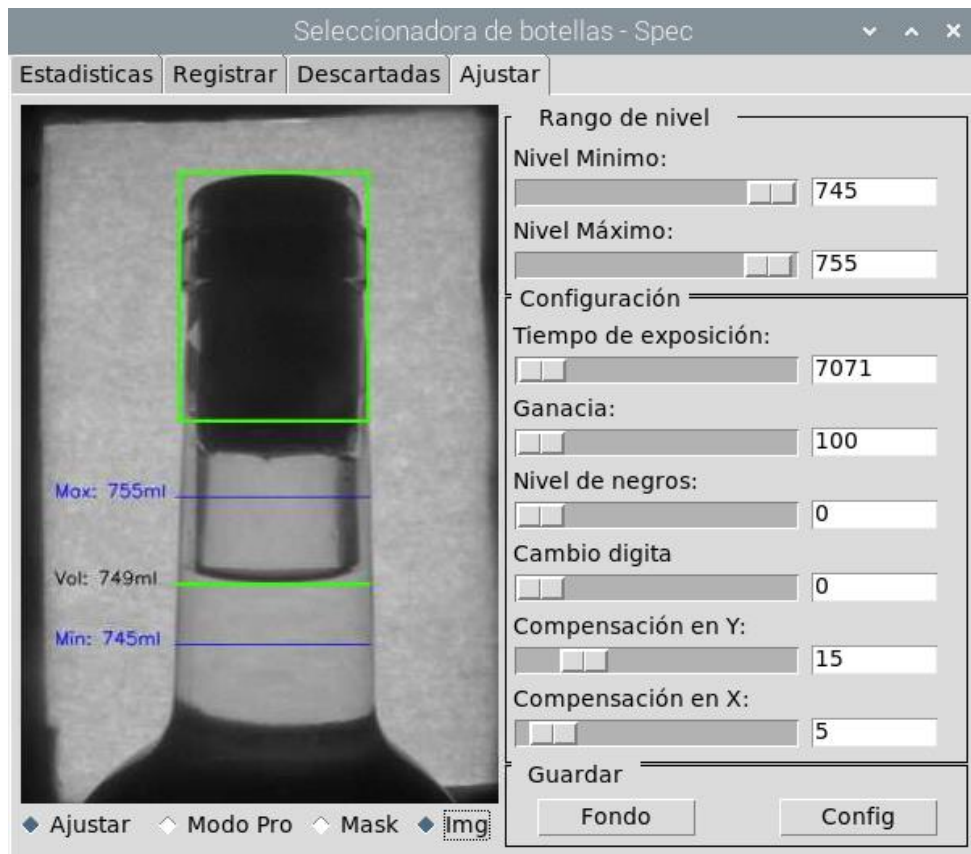


Ilustración 3.17 - Pestaña Ajustar, botella OK

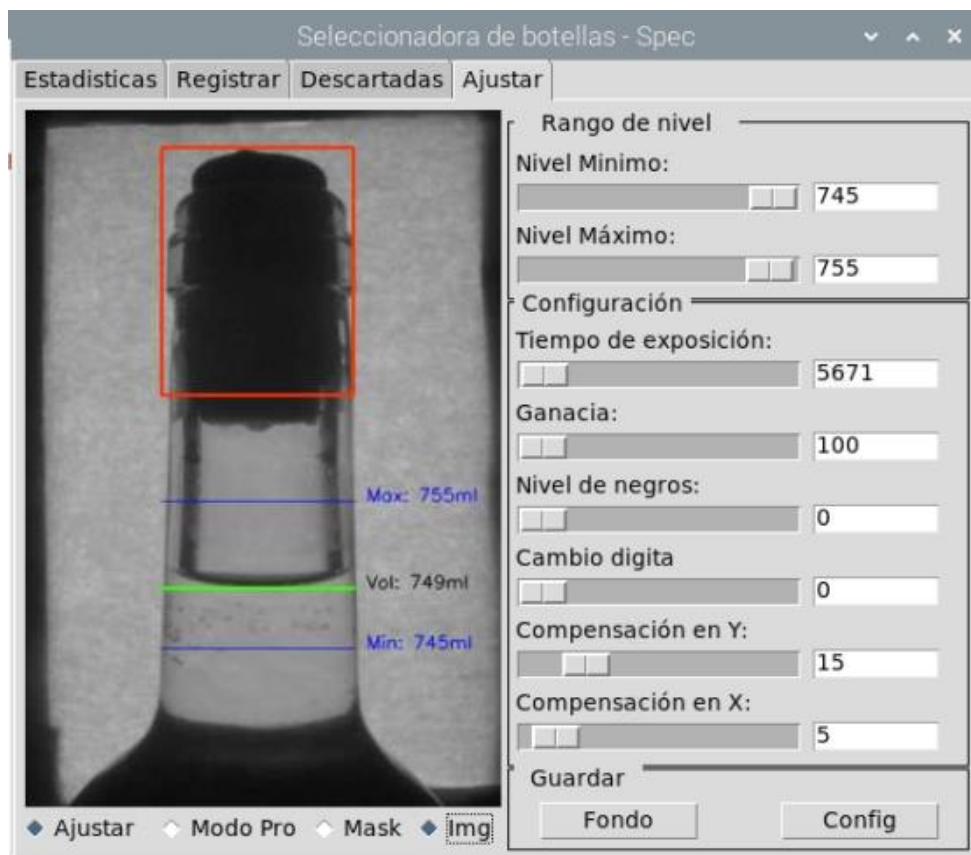


Ilustración 3.18 - Pestaña Ajustar, botella NOK

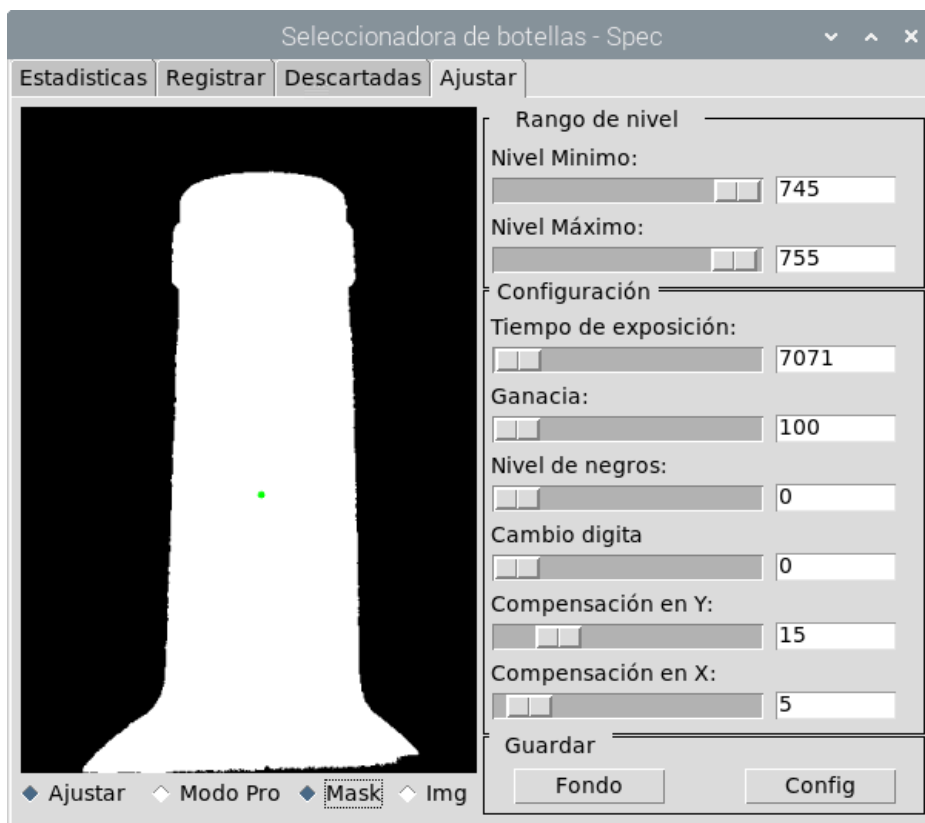


Ilustración 3.19 - Pestaña Ajustar, máscara definitiva

### 3.3.3.1.6 Modo operador

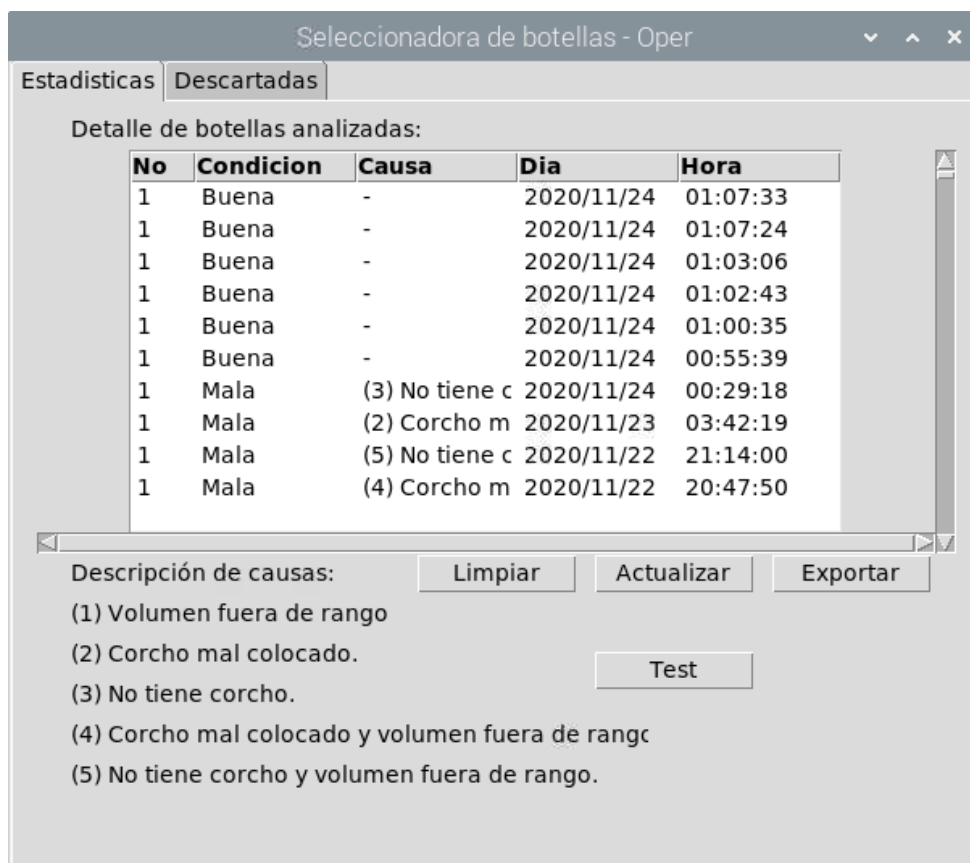


Ilustración 3.20 - Ventana del modo Operador

Como se puede observar el modo operador solo da acceso a las pestañas Estadísticas y Descartadas, las cuales funcionan igual que en el modo que en la ventana del especialista.

### 3.3.3.2 Adquisición y Análisis de imágenes

A continuación, se detalla el procedimiento utilizado para la adquisición de imágenes y el procesamiento y análisis de estas, indicando los factores físicos e informáticos que intervienen en este proceso.

Para llevar a cabo estas tareas se elaboró un prototipo basado en las medidas del 3.3.4.2 Diseño del prototipo.

#### 3.3.3.2.1 Adquisición de imágenes

La adquisición de imágenes se puede dividir en tres partes:

- Configuración inicial de la cámara
- Ajuste de iluminación y enfoque
- Adquisición

##### 3.3.3.2.1.a Configuración inicial de la cámara

El primer parametro a configurar en la camara es su IP, se debe colocar la en la misma red en que esta el ordenador. Para conseguirlo se ha utilizado el software del fabricante “*pylon IP Configurator*”, este permite ver la IP y modificar la IP de la camara.

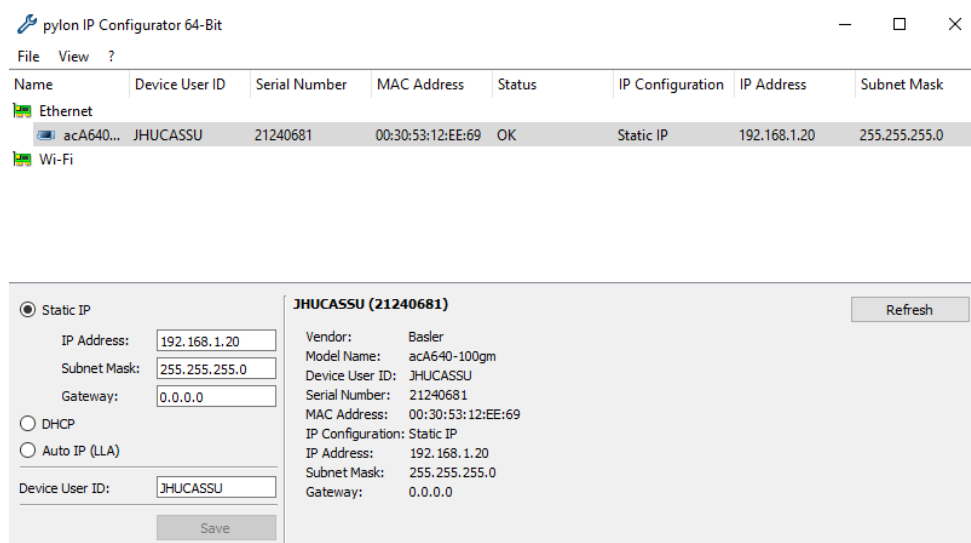


Ilustración 3.21 - Configuración inicial, pylon IP Configurator

Una vez se ha configurado la IP con el software del fabricante “*pylon viewer*”, se da la configuración inicial a la cámara para que pueda mostrar las imágenes y tomar las fotos en el formato que se necesita para la aplicación. Los parámetros que se han configurado son:

No.	Parámetro	Valor
<b>Image Format Controls</b>		
1	Píxel Format	YUV 422 Packed
<b>AOI Controls</b>		
2	Width	580 px
3	Height	400 px
<b>Adquisition Controls</b>		
4	Trigger Mode	Off
<b>Transport Layer</b>		

<b>5</b>	<b>Packet Size</b>	<b>1500 bytes</b>
----------	--------------------	-------------------

Tabla 3.7 - Parámetros de configuración inicial de la cámara

### 3.3.3.2.1.b Ajuste de iluminación y enfoque

Ver el punto 3.3.2.4 Iluminación para más detalles.

El nivel de iluminación, los lúmenes, de la backlight deben ser superiores a los aportados por la luz “parasita” del entorno. Esto para poder cerrar el diafragma de la cámara hasta el punto de que no capte la iluminación exterior y la imagen este bien expuesta solo con la backligh, a la velocidad de obturación requerida.

Una vez se tiene la iluminación cerca de los parámetros requeridos y con el diafragma lo más cerrado posible (con esto se obtiene una mayor profundidad de campo), se hace uso de la herramienta “Sharpness indicator” del software *pylon Viewer* del fabricante.

La herramienta de enfoque del fabricante indica que tan enfocada esta la imagen, según se gira el anillo de enfoque del lente. Es suficiente llevar el enfoque a un 90%.

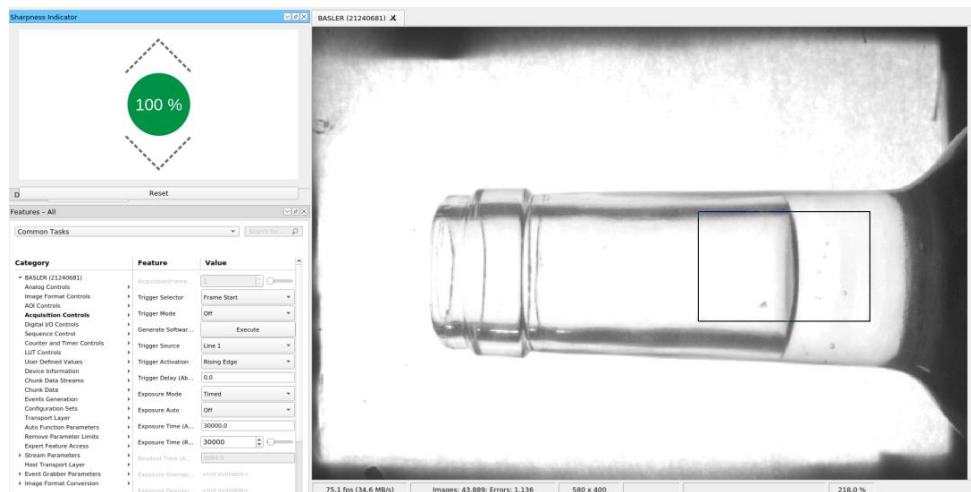


Ilustración 3.22 - Enfoque de imagen, pylon viewer

### 3.3.3.2.1.c Adquisición

La señal para disparar la cámara la da el interruptor infrarrojo, que censa cuando una botella va a pasar por el área de captura de la cámara. Esta señal solo es recibida cuando la sesión de uno de los usuarios está activa.

Una vez el sistema recibe la señal, se ejecuta la función “SaveImage(path)” la cual dispara la cámara con el comando “camara.GrabOne(5000)” y procesa el resultado y lo guarda con la función `cv2.imwrite(path,img)`.

### 3.3.3.2.2 Procesamiento y análisis de imágenes

El procesamiento y análisis de las imágenes se basa en los siguientes puntos:

- Definir el área de interés, en este caso definir la ubicación de la botella en la imagen.
- Determinar el nivel de líquido.
- Determinar la presencia y posición del tapón.
- Comparar los valores obtenidos con los admisibles.

Para hacer esta tarea se han utilizado de las librerías OpenCV y numpy en la elaboración de varias funciones que permiten lograr este cometido a la perfección.

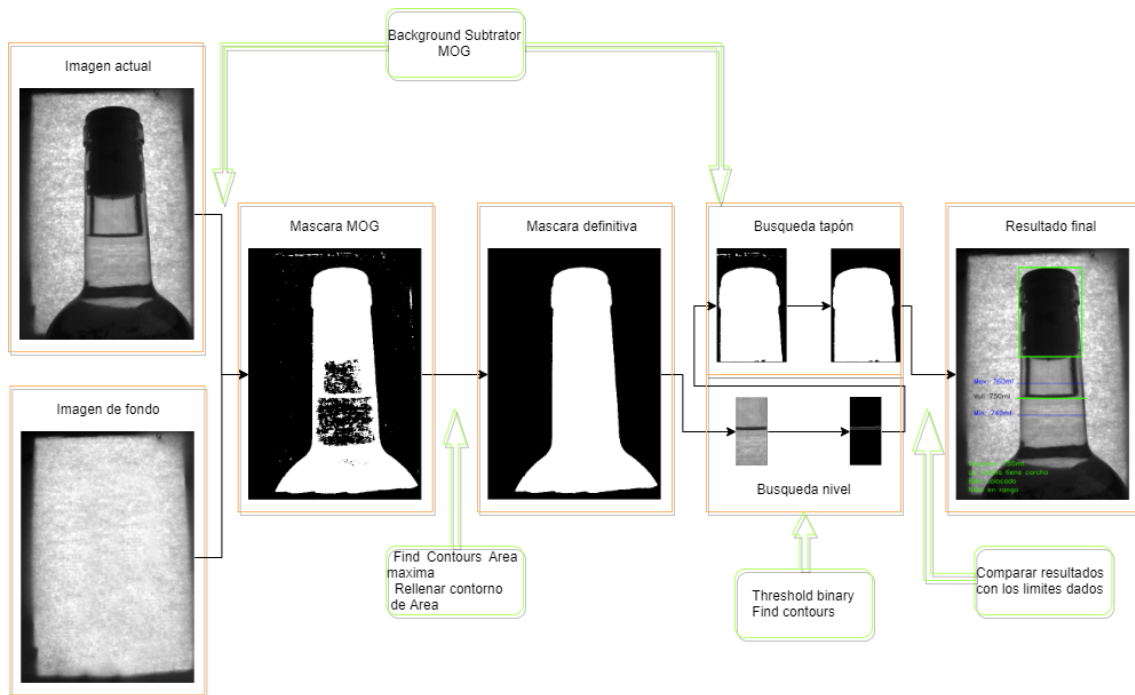


Ilustración 3.23 - Esquema de proceso de análisis de imagen

### 3.3.3.2.2.a Definir área de interés

El propósito de esta etapa es distinguir el objeto (botella) del fondo de la imagen para poder enfocarnos en estos píxeles que serán nuestros píxeles de interés.

Para lograr este cometido lo primero que se hace es crear un objeto “BackgroundSubtractorMOG” el cual se aplica a una secuencia de imágenes formadas por la imagen del fondo y la nueva imagen tomada por la cámara.

### 3.3.3.2.2.b Determinar nivel de líquido

El nivel de líquido de la botella se delimita por las variables “max\_level\_img\_px” y “min\_level\_img\_pixel” y por la variable “bottle\_width”. A partir de estos valores se hace una ROI para que el nivel sea buscado en el área donde se entiende que debería estar. Esto permite acotar el área de búsqueda y reducir el número de falsos positivos.

A esta ROI se le aplica un Threshold binary y se aplica una búsqueda de contornos y se aplica la función “cv2.boundingRect(contorno)” y se toma el mayor de todos los contornos que tuvieron una anchura mayor al 60% del ancho de la botella. Este valor es el que se asume como nivel de líquido.

Una vez se tiene el nivel de líquido en píxeles, se calcula en que porcentaje de la altura de la imagen se encuentra y este valor es convertido a mililitros. Esta conversión se hace mediante un mapeo elaborado con diferentes medidas.

Para la elaboración del mapeo, se partió de que la densidad del agua es aproximadamente 1000 kg/m<sup>3</sup>, por lo que cada gramo de agua equivale a un mililitro de esta. Se pesó la botella con agua con tres medidas distintas 745, 750 y 750 gr.

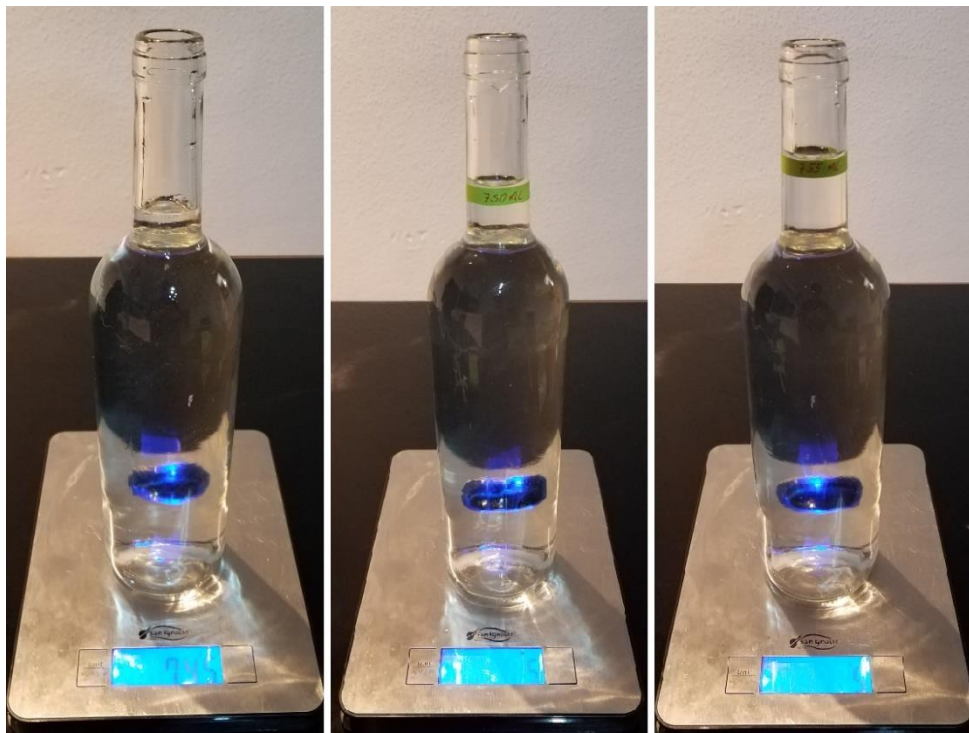


Ilustración 3.24 - Medidas de volumen para mapeo

Cada una de las medidas fue colocada frente a la cámara para obtener porcentaje de la imagen que ocupaba el nivel, en donde 0.00 es la base de la imagen y 1.00 la parte superior.

De este análisis obtuvimos los siguientes valores:

Volumen	Equivalente
745 ml	0,228
750 ml	0,333
755 ml	0,438

Tabla 3.8 - Equivalencia de volumen

El resto de los valores del mapa fueron obtenidos asumiendo que el cuello de la botella es un cilindro uniforme, por lo que la relación entre la altura y el volumen tienen una relación proporcional directa y, por tanto:

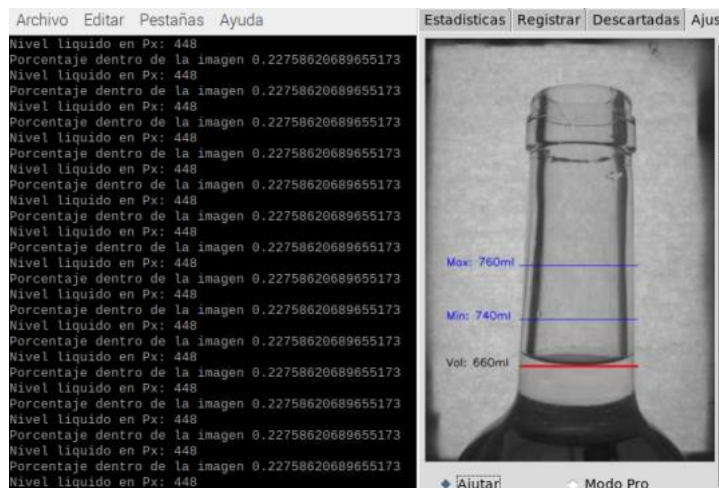


Ilustración 3.25 - Obtención de equivalente de volumen

$$Volumen = 47,619 * Equivalente + 734,14$$

### 3.3.3.2.2.c Determinar la presencia y posición del tapón

La determinación de la presencia del sensor se hace aplicando el mismo método del Background Subtrator MOG. Se aplica la búsqueda de contornos y se toma la de mayor valor para compararla con una constante preestablecida, si es menor a este valor se establece que no hay tapón.

### 3.3.3.2.2.d Comparar los valores obtenidos con los admisibles

En cada una de las etapas antes mencionadas se almacenan los valores de interés en variables para para luego ser evaluados en la etapa de representación gráfica. Estas variables se comparan con los límites preestablecidos para el nivel y las dimensiones de las botellas.

En el caso del nivel, se compara el valor obtenido con el límite mínimo y máximo del nivel. Si el nivel está dentro del rango establecido se marca en la imagen de salida con una línea color verde y si esta fuera con una línea color roja, además se le envía la información a la parte de estadísticas para que sea registrado.

En el caso del corcho, se hace una comparación de áreas para determinar la presencia de este. Si el área obtenida es menor al límite de área establecido se indica que no hay corcho, en caso contrario, presencia de corcho, se evalúa la parte más elevada de la imagen tomada y se compara con el límite preestablecido, si el valor obtenido está por encima al límite se dibuja un rectángulo color rojo y se indica que el corcho está mal colocado. Si el valor está por debajo del límite se dibuja el rectángulo de color verde.

## 3.3.4 Ambiente físico

La parte físicas se puede dividir en dos partes, la interconexión de los equipos y el diseño mecánico del prototipo.

### 3.3.4.1 Interconexión de equipos

La conexión de los componentes se puede dividir en la conexión de red y en la conexión eléctrica. En la imagen se muestra en verde la conexión Ethernet y en rojo conexión PoE.

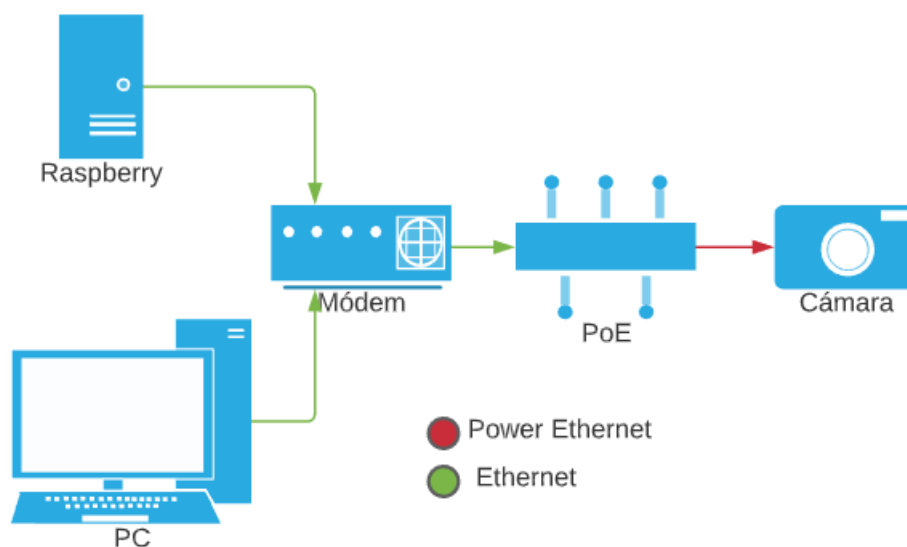


Ilustración 3.26 - Esquema de conexión Ethernet y PoE

Para la interconexión del interruptor infrarrojo y la Raspberry pi se hizo uso de un divisor de tensión, ya que la alimentación y salida del sensor es a 5V y la Raspberry solo soporta 3V en las entradas.

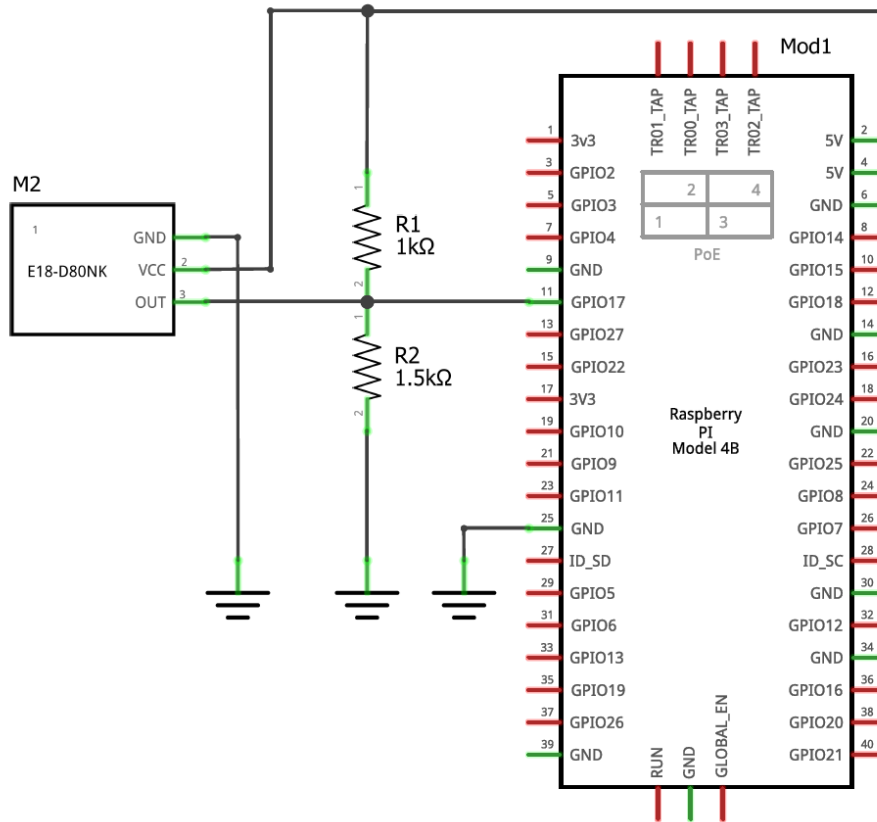


Ilustración 3.27 - Esquema de conexión eléctrica

También se muestra la conexión física de los dispositivos.

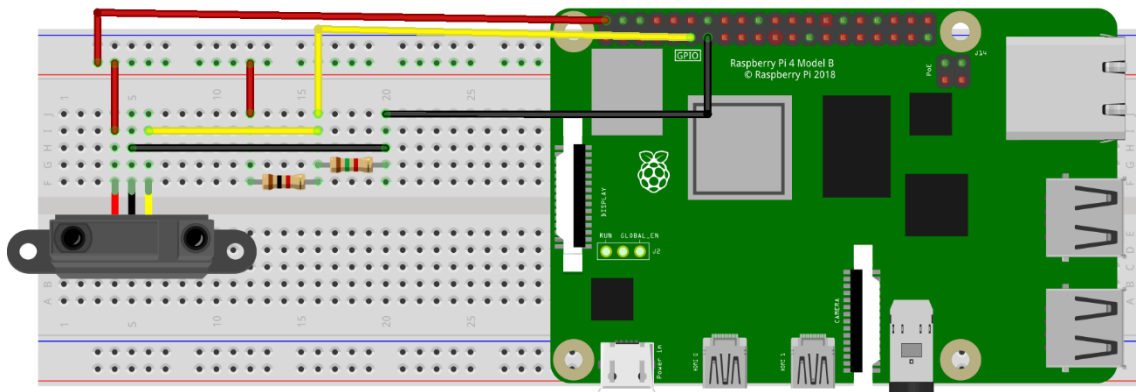


Ilustración 3.28 - Conexión en protoboard



### 3.3.4.2 Diseño de prototipo

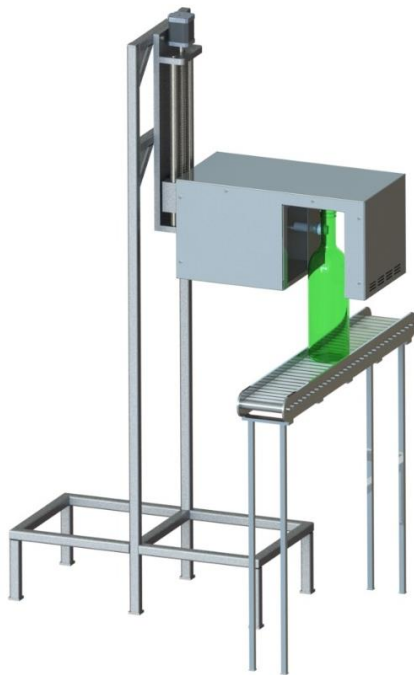


Ilustración 3.29 - Prototipo diseñado

Las medidas del prototipo se establecieron después de haber seleccionados los equipos principales a ser utilizados: cámara, luz y lente.

- Se buscó el lente con la menor distancia de enfoque posible (10 mm) y que fuera de una distancia focal pequeña (4,5 mm – Gran angular) para que el prototipo fuera lo más pequeño posible.

Luego se calculó a que distancia de enfoque se tomaba toda la pantalla dentro del frame de la cámara. Sabiendo que el sensor de cámara es de ¼" (3,60mm de ancho)

- Se seleccionó una luz industrial con las dimensiones en función del objeto que se iba a iluminar. El área útil de la luz es de 150x100.

Una vez obtenidas todas las dimensiones principales se inició el proceso de diseño.

Todas las partes de la máquina fueron diseñadas con SOLIDWORKS Education Edition 2019, el módulo de Sistemas de Láminas Metálicas y Estructuras fue usado

en el diseño.

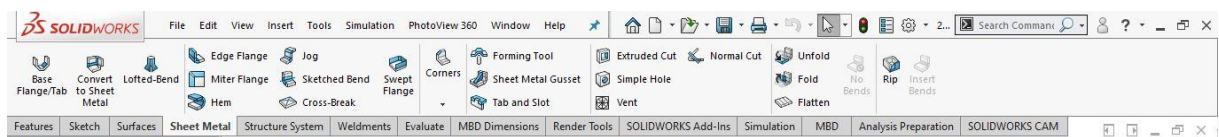


Ilustración 3.30 - Modulo de SolidWorks - Sheet Metal

Diseñando con chapa metálica hay algunos parámetros que son obligatorios para el proceso de diseño. Uno de los parámetros de la chapa metálica es el espesor de esta. El grosor de la chapa es importante para soportar todas las cargas que las otras partes de la máquina se imponen unas a otras. Para considerar este parámetro, todo el espesor de la chapa se toma 2 mm. Además, el material a seleccionar es también un elemento importante para soportar las cargas aplicadas a lo largo del proceso de trabajo de la máquina. El material seleccionado es el acero inoxidable al cromo. La siguiente tabla muestra las propiedades del material.

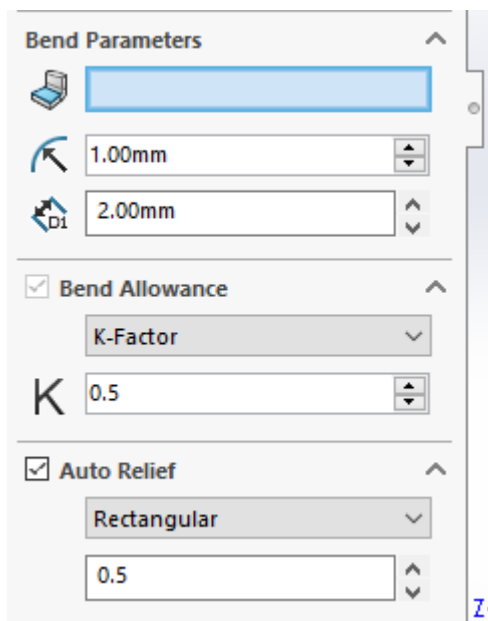


Ilustración 3.31 - parámetros de doblado de la chapa

Propiedad	Valor	Unidad
Módulo elástico	2e+11	N/m <sup>2</sup>
Relación de Poisson	0.28	NO.
Módulo de cizalla	7.7e+10	N/m <sup>2</sup>
Densidad de la masa	7800	kg/m <sup>3</sup>
Resistencia a la tensión	413613000	N/m <sup>2</sup>
Fuerza de compresión		N/m <sup>2</sup>
Fuerza de rendimiento	172339000	N/m <sup>2</sup>
Coefficiente de expansión térmica	1.1e-05	/K
Conductividad térmica	18	W/(m-K)
Calor específico	460	J/(Kg.K)
Ratio de amortiguación del material		N/A

Tabla 3.9 - Características de material utilizado

El comando de la brida de la base durante el proceso de diseño se utiliza para obtener una pieza angular de chapa. Para ello, Solidworks necesita algunos de los parámetros que son el radio de curvatura y el Factor K. A estos dos parámetros se les da la tabla.

Radio de curvatura	1 mm
Factor K	0,5 mm

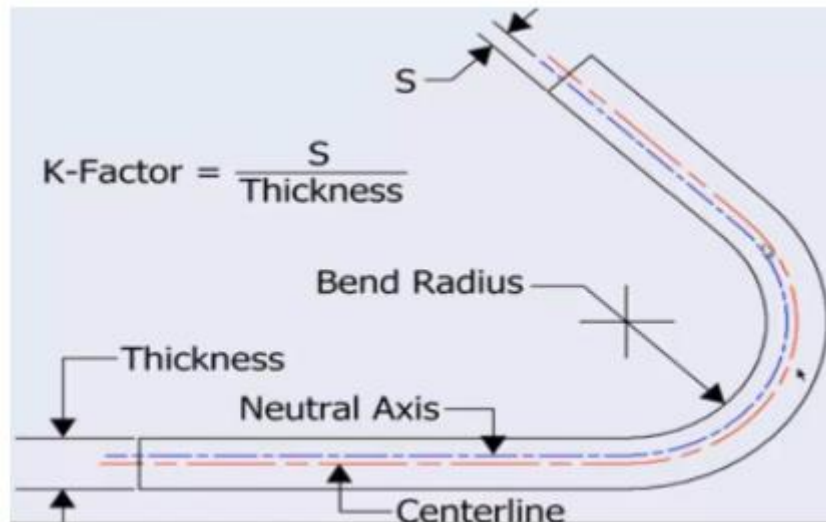


Ilustración 3.32 - Doblado de chapa

$$K - Factor = \frac{S}{Thickness} \quad (1)$$

$$0,5 = \frac{S}{2} \text{ and } S = 1 \text{ mm} \quad (2)$$

Después de especificar estos parámetros en la sección de material y en el módulo de chapa, en el menú superior de Solidworks se puede ver el módulo de chapa que se utiliza como parte principal de la máquina.

Mecanismo estructural de la pierna diseñado para soportar todo el peso de la máquina y mantener la máquina a cierta altura. Para este diseño se utiliza el Sistema Estructural. Todo el perfil es de 20X20X2.

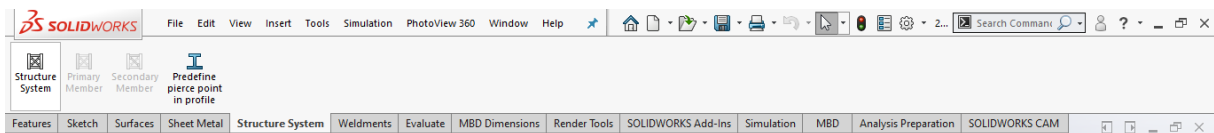


Ilustración 3.33 - Modulo SolidWorks, Structure System

### 3.3.5 Análisis de costos

El costo del prototipo se divide en tres partes, en el costo de las estructuras diseñadas, el costo de los dispositivos adicionales como cámara, Raspberry y demás, y en las horas de ingeniería requeridas para la integración y configuración de la máquina.

#### 3.3.5.1 Costo de dispositivos

En la Tabla 3.10 se muestra un listado con el precio de los dispositivos requeridos para el ensamblaje del prototipo.

No.	Descripción	Cantidad	Unidad	Precio	Total
1	Cámara Basler ACA640-100gm	1	UN	373,49 €	373,49 €
2	MVL5WA - 4.5 mm EFL, f/1.4, for 1/2" C-Mount Format Cameras, with Lock	1	UN	173,13 €	173,13 €

3	Backlight BKL1510A-W00C	1	UN	983,99 €	983,99 €
4	Raspberry Pi Spain RAS-4-4G - Placa Base Pi 4 Modelo B / 4 GB SDRAM (1822096)	1	UN	63,85 €	63,85 €
5	SanDisk Ultra Tarjeta de memoria microSDHC con adaptador SD, hasta 98 MB/s, rendimiento de apps A1, Clase 10, U1, 32 GB	1	UN	5,90 €	5,90 €
6	TP-Link TL-PoE150S Gigabit Ethernet PoE Inyector, Plug-and-Play, 2 puertos 10/100/1000Mbps RJ45, 1 puerto de 48VDC, 15,4 W (máximo 48 V DC)	1	UN	18,36 €	18,36 €
7	NanoCable 10.20.0401 - Cable de red Ethernet RJ45 Cat.6 UTP AWG24, 100% cobre, Gris, latiguillo de 1mts	4	UN	1,75 €	7,00 €
8	E18-D80NK Interruptor de sensor de infrarrojos para evitar Interruptor de sensor de detección de obstáculos de infrarrojos ajustable	1	UN	11,89 €	11,89 €
9	AABCOOLING Black Silent Fan 8 - Un Silencioso y Muy Efectivo Ventilador PC, Fan 80mm, Ventilador 12V, Ventilador Laptop, Fan Cooler, 50 m3/h, 2000 RPM 15,6 dB (A)	1	UN	6,99 €	6,99 €
10	INOX 2000-1000-2 PLANCHA A-304	32	KG	7,65 €	244,67 €
11	INOX 20-20-1 TUBO AISI-304	4,2	MT	8,68 €	36,44 €
				<b>TOTAL:</b>	<b>1.925,71 €</b>

Tabla 3.10 - Costo de dispositivos

### 3.3.5.2 Costo de estructura diseñada

Para el cálculo del coste de la estructura diseñada del prototipo se hizo uso de la herramienta "Costing" de SolidWorks. En esta se configuraron las plantillas de planchas de metal (Sheet metal template) y la de mecanizado (Machining template) con los precios correspondientes a los materiales utilizados (Ver anexo 4).

En la Tabla 3.11 se muestran los resultados obtenidos.

No.	Componentes	Coste de material	Coste de fabricación	Coste total
1	2020_Machine_001_Linear_Act	59,26 €	123,62 €	182,88 €
2	2020_Machine_002_Leg	42,10 €	84,54 €	126,64 €
3	2020_Machine_001_Box	19,58 €	58,32 €	77,90 €

4	2020_Machine_001_Box_Front	9,28 €	22,01 €	31,29 €
5	2020_Machine_001_Box_Back	9,28 €	21,88 €	31,16 €
6	SFU1605-1610-1	1,95 €	21,29 €	23,25 €
7	2020_Machine_001_Linear_Act_Connect	7,13 €	12,73 €	19,86 €
8	2020_Machine_001_Light-Camera	8,52 €	9,81 €	18,32 €
9	SFU1605-1610-3	0,12 €	11,05 €	11,17 €
10	SFU1605-1610-2	0,12 €	11,05 €	11,17 €
	<b>Total</b>	<b>157,34 €</b>	<b>376,30 €</b>	<b>533,65 €</b>

Tabla 3.11 - Costo de estructura diseñada

### 3.3.5.3 Costo total

Al agregar el costo de las horas de ingeniería requeridas para montar los dispositivos en la estructura diseñada y configurar los parámetros iniciales correspondientes a la forma y volumen de la botella. Se estima que se requieren unas 4 horas de trabajo, aun costo de 8,50 € por hora.

Tomando en consideración 21% del impuesto sobre el valor añadido (IVA).

No.	Descripción	Costo	IVA	Costo Total
1	Costo de dispositivos	1.925,71 €	- <sup>4</sup>	1.925,71 €
2	Costo de estructura	533,65 €	112,07 €	645,72 €
3	Configuración	34,00 €	7,14 €	41,14 €
	<b>Total</b>	<b>2.493,36 €</b>	<b>523,61 €</b>	<b>2612,56 €</b>

Tabla 3.12 - Costo total sin ingeniería ni desarrollo

El costo de la Tabla 3.12 no contempla el costo del de desarrollo de la aplicación. Asumiendo que el desarrollo de una aplicación, de este tipo, para una persona experimentada de 110 horas a un costo de 12,50 € por hora, obtenemos el costo de desarrollo siguiente:

No.	Descripción	Costo	IVA	Costo Total
1	Ingeniería y desarrollo	1.375,00 €	288,75 €	1.663,75 €
	<b>Total</b>	<b>1.375,00 €</b>	<b>288,75 €</b>	<b>1.663,75 €</b>

Tabla 3.13 - Costo de la ingeniería y el desarrollo

<sup>4</sup> IVA incluido en costo.

## 4 Conclusiones

Este proyecto, con el cual se ha desarrollado un prototipo que replica las funcionalidades de máquinas de control de calidad industriales, ha dado demostrar que las librerías Open source, como OpenCV, de procesamiento y análisis de imágenes tienen una fiabilidad aceptable, dentro de unos límites iluminación establecidos. Permitiendo desarrollar aplicaciones que puedan funcionar en microcontroladores de bajo coste, en comparación con un ordenador, como la Raspberry pi 4B sin problemas.

De acuerdo con los resultados obtenidos la Raspberry es capaz de controlar una cámara industrial a la vez que ejecuta algoritmos de reconocimiento de objetos capaces de detectar distintas formas y objetos, y que OpenCV aun siendo librería open source, gratuita, da buenos resultados en tarea de procesamiento y análisis de imágenes. Brindando a los usuarios números filtros y funciones que bajo las condiciones externas adecuadas pueden ayudar a cumplir muchas de las tareas que se requieren en la industria.

Sin embargo, el que se hayan tenido buenos resultados en el desarrollo de la aplicación no es suficiente para llevar esta tarjeta a la hostilidad del mundo industrial. Aun la Raspberry da buenos resultados se requiere de equipos periféricos que ayuden a aumentar su confiabilidad, a la vez que se lleven a cabo pruebas y ensayos que pongan a prueba sus capacidades en conjunto con estos periféricos.

Python y OpenCV en conjunto brindan una gran gama de posibilidades a la hora de desarrollar softwares basados en la visión artificial, permitiendo crear aplicaciones de fácil configuración y manejo para el usuario de bajo nivel de conocimiento en el área.

En lo referente al diseño de las partes físicas del sistema SolidWorks ha demostrado ser una herramienta muy poderosa a la hora de desarrollar prototipos, dando la posibilidad elaborar presupuestos de manera fácil, sencilla y versátil. Permitiendo al usuario evaluar la variación en los costos producidos por cambios de material y/o cantidades.

## 5 Bibliografía

- B, D. (31 de Agosto de 2019). *towards data science*. Obtenido de <https://towardsdatascience.com/extracting-regions-of-interest-from-images-dacfd05a41ba>
- Basler AG. (2020). *Basler*. Obtenido de Downloads Software: <https://www.baslerweb.com/en/sales-support/downloads/software-downloads/>
- Basler AG. (30 de Enero de 2020). PYLON DEPLOYMENT GUIDE. Ahrensburg, Germany.
- Cano, J. A. (Enero de 2010). Diseño y Fabricación de una Pila de Combustible de Hidrógeno de Baja Potencia. Catalunya, España. Obtenido de <https://upcommons.upc.edu/bitstream/handle/2099.1/8939/pliego.pdf>
- dano. (9 de Septiembre de 2014). *Stackoverflow*. Obtenido de <https://stackoverflow.com/questions/25753632/tkinter-how-to-use-after-method>
- DCM SISTEMES. (2020). *DCM Sistemes*. Retrieved from TECHNICAL DATA, ADDITIONAL INFORMATION & DOWNLOADS: <https://www.dcmsistemas.com/en/technical-data/ibluedriver/97/ibluedriverbox/>
- Emmet. (19 de diciembre de 2019). *PiMyLifeUp*. Obtenido de Installing OpenCV on the Raspberry Pi: <https://pimylifeup.com/raspberry-pi-opencv/>
- Hernández, L. d. (17 de Mayo de 2018). *programarfacil.com*. Obtenido de <https://programarfacil.com/blog/vision-artificial/detector-de-bordes-canny-opencv/>
- it-swarm.dev. (31 de Marzo de 2020). *it-swarm.dev*. Obtenido de <https://www.it-swarm.dev/es/python/lea-una-imagen-con-opencv-y-muestrela-con-tkinter/1051338993/>
- Kasalický, P. (20 de Septiembre de 2019). *github.com*. Obtenido de mbalatsko/pypylon-opencv-viewer: <https://github.com/mbalatsko/pypylon-opencv-viewer/blob/master/README.md>
- MNP RasPi Blog 2016. (27 de Julio de 2016). *mnp*. Obtenido de <https://www.mnp.cl/post/ques-board-bcm-raspberry-pi>
- Open Source Computer Vision. (30 de Noviembre de 2020). *OpenCV*. Obtenido de [https://docs.opencv.org/3.4/d1/dc5/tutorial\\_background\\_subtraction.html](https://docs.opencv.org/3.4/d1/dc5/tutorial_background_subtraction.html)
- OpenCV-Python Tutorials. (2020). *OpenCV-Python Tutorials*. Obtenido de Contour Properties: [https://opencv-python-tutroals.readthedocs.io/en/latest/py\\_tutorials/py\\_imgproc/py\\_contours/py\\_contour\\_properties/py\\_contour\\_properties.html](https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_imgproc/py_contours/py_contour_properties/py_contour_properties.html)
- Paul. (Abril de 21 de 2018). *Solarian Programmer*. Obtenido de <https://solarianprogrammer.com/2018/04/21/python-opencv-show-video-tkinter-window/>
- Pesotskiy, V. (7 de Diciembre de 2018). *Information Service Systems*. Obtenido de <https://blog.issart.com/opencv-to-line-production/>
- Raspberry pi Foundation. (2020). *raspberrypi.org*. Obtenido de GPIO: <https://www.raspberrypi.org/documentation/usage/gpio/>
- Raspberry Pi Trading Ltd. (Mayo de 2020). Raspberry Pi 4 Computer Model B.

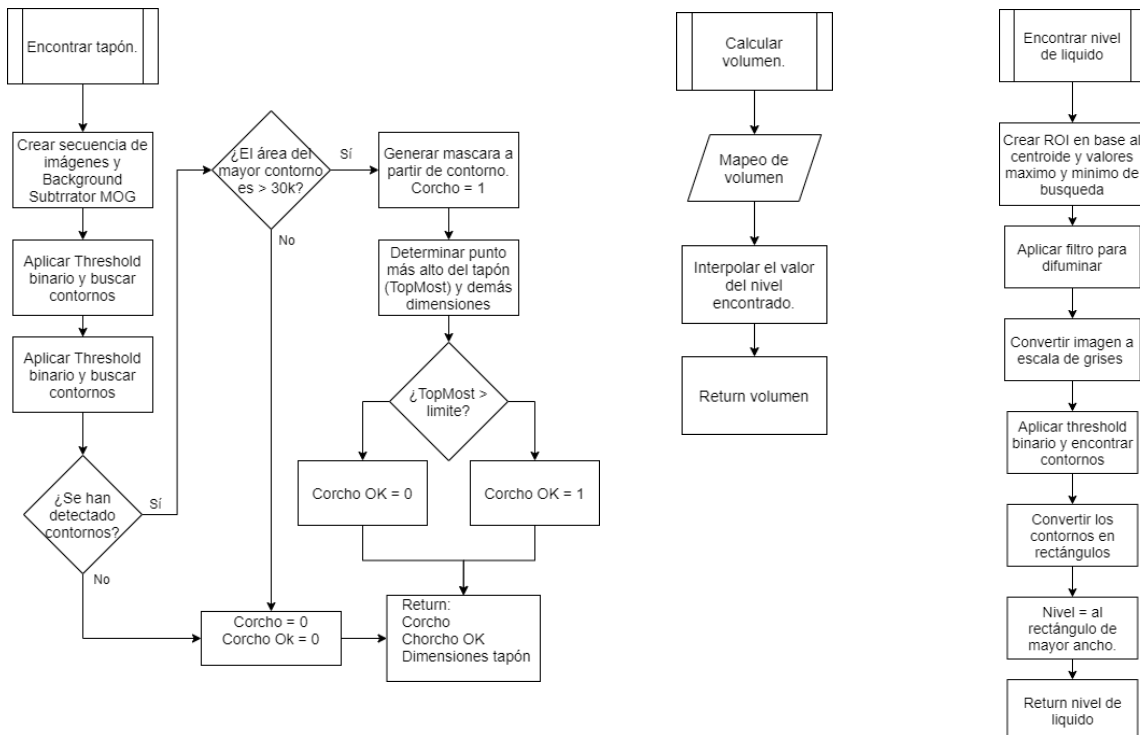
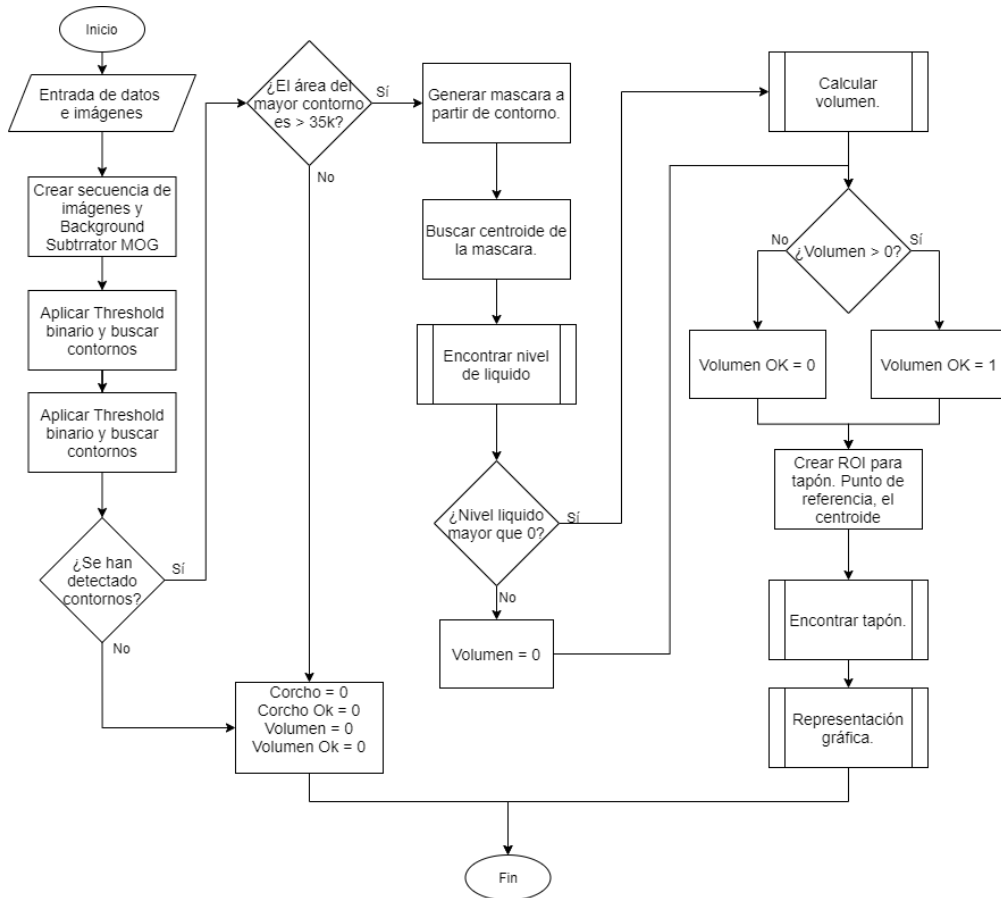
- raspberrypi.org. (29 de Junio de 2016). *raspberrypi.org*. Obtenido de Avoiding False Hits with RPi.GPIO Edge Detection:  
<https://www.raspberrypi.org/forums/viewtopic.php?t=134394>
- Raspi.TV. (25 de Julio de 2014). *Raspi.TV*. Obtenido de RPi.GPIO update and detecting BOTH rising and falling edges: <https://raspi.tv/2014/rpi-gpio-update-and-detecting-both-rising-and-falling-edges>
- rishikagupta1. (19 de 02 de 2020). *GeeksforGeeks*. Obtenido de <https://www.geeksforgeeks.org/background-subtraction-opencv/>
- Soloriam Programer. (20 de Abril de 2018). *My programming ramblings*. Obtenido de Python OpenCV - show an image in a Tkinter window:  
<https://solarianprogrammer.com/2018/04/20/python-opencv-show-image-tkinter-window/>
- Sourcecodester. (26 de Julio de 2018). *Sourcecodester.com*. Obtenido de Python - Import CSV File To Tkinter Table:  
<https://www.sourcecodester.com/tutorials/python/12494/python-import-csv-file-tkinter-table.html>



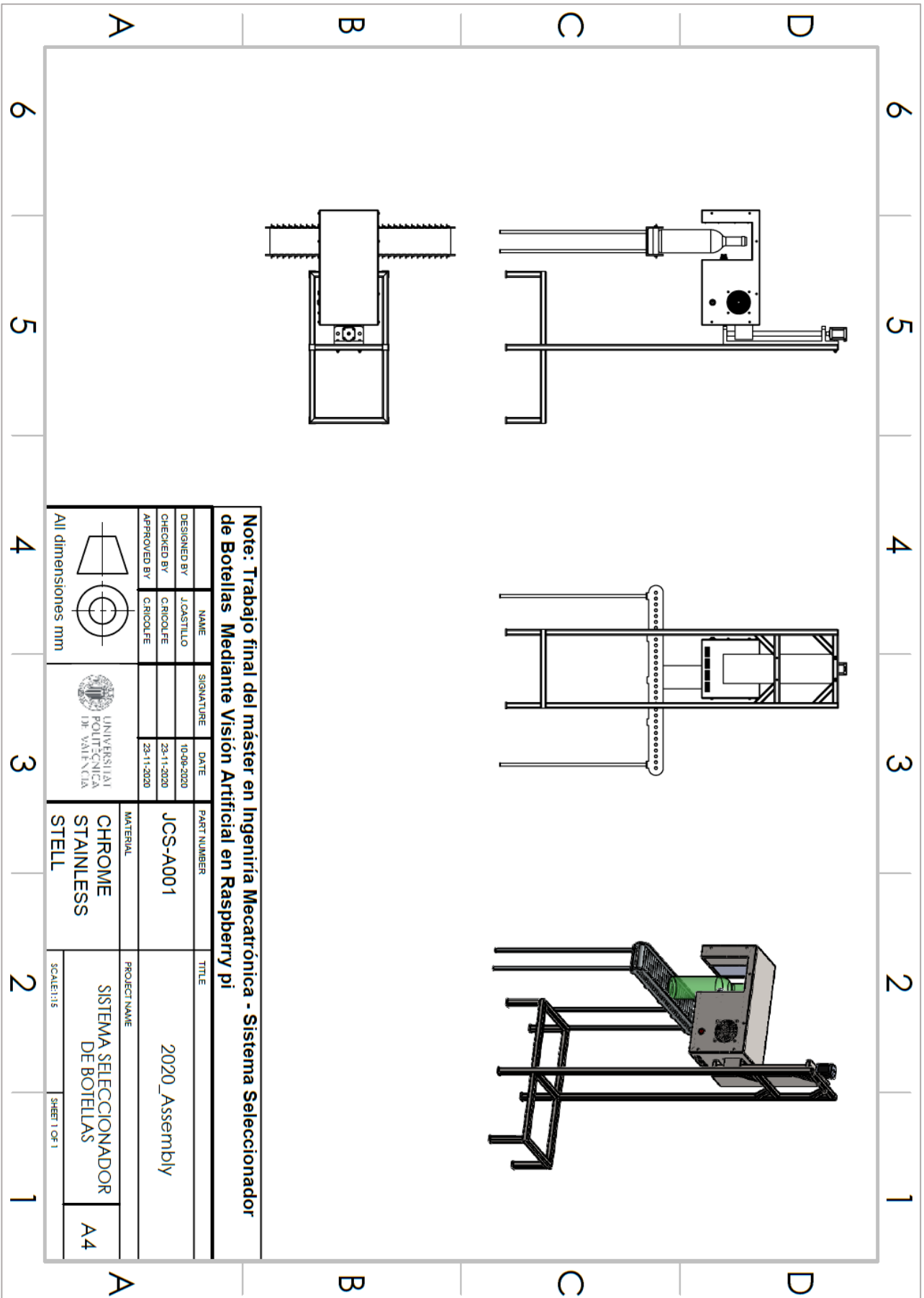
# Anexos

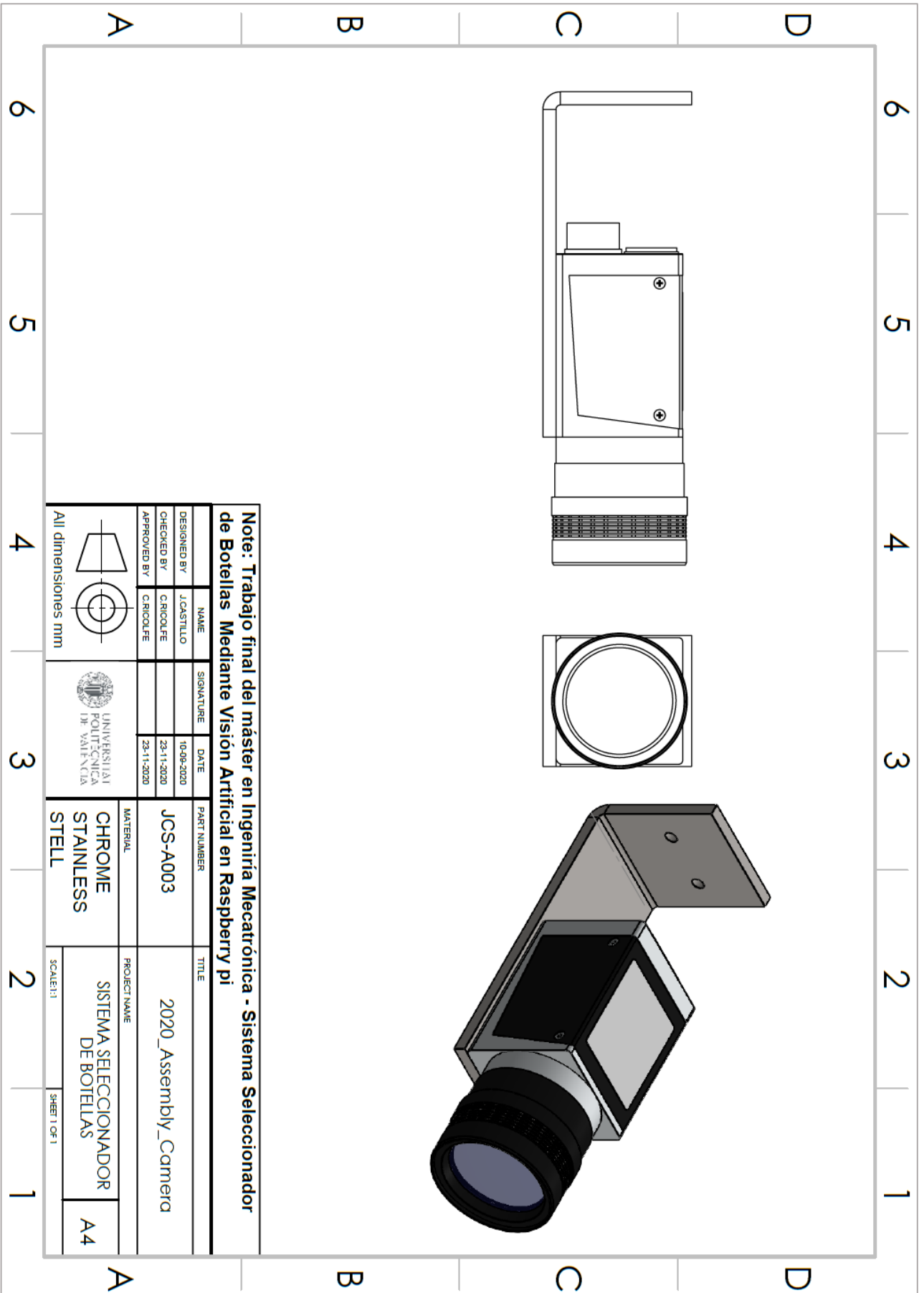
## Anexo 1 - Diagramas de flujo

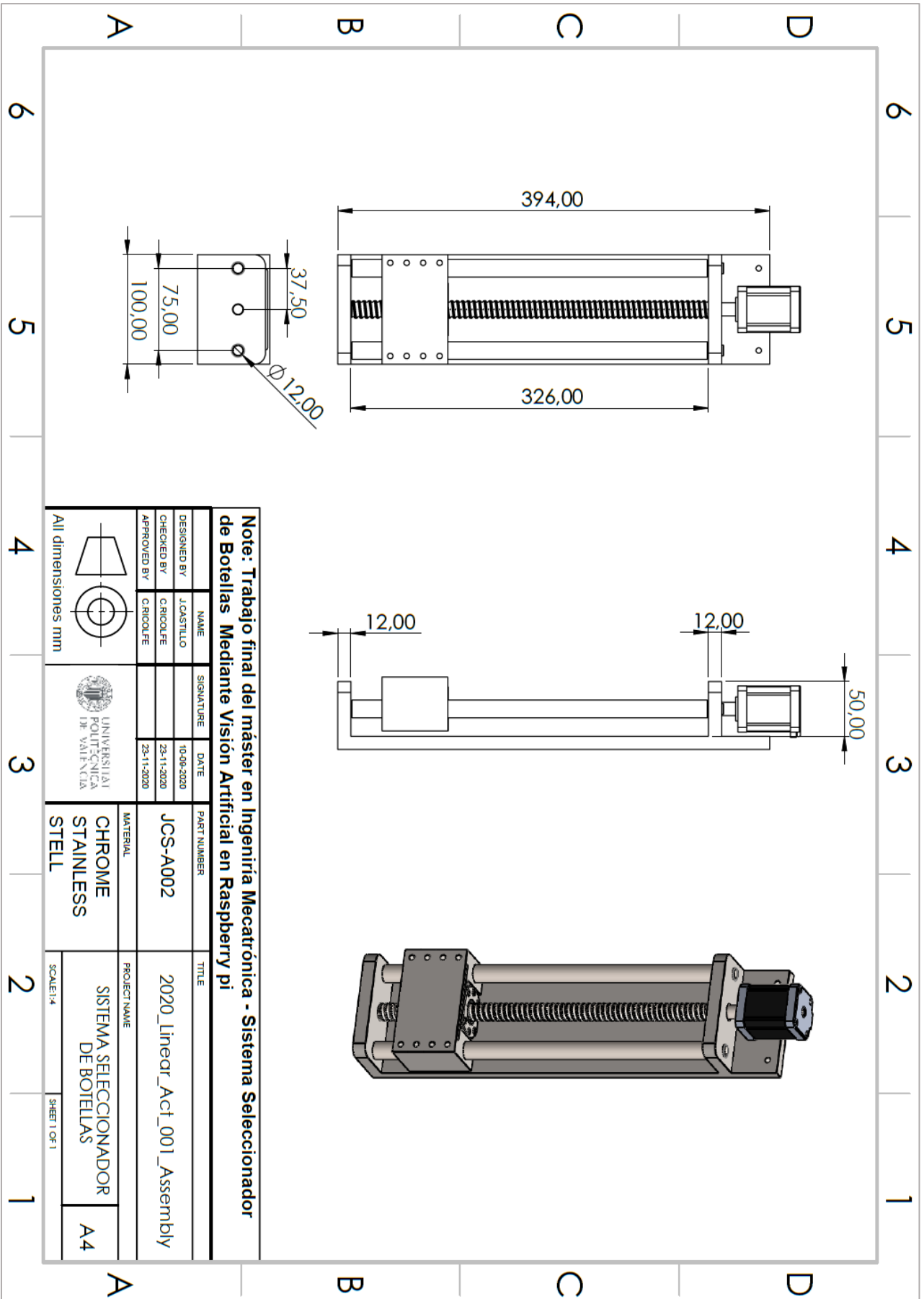
Diagrama de flujo del análisis de imágenes.

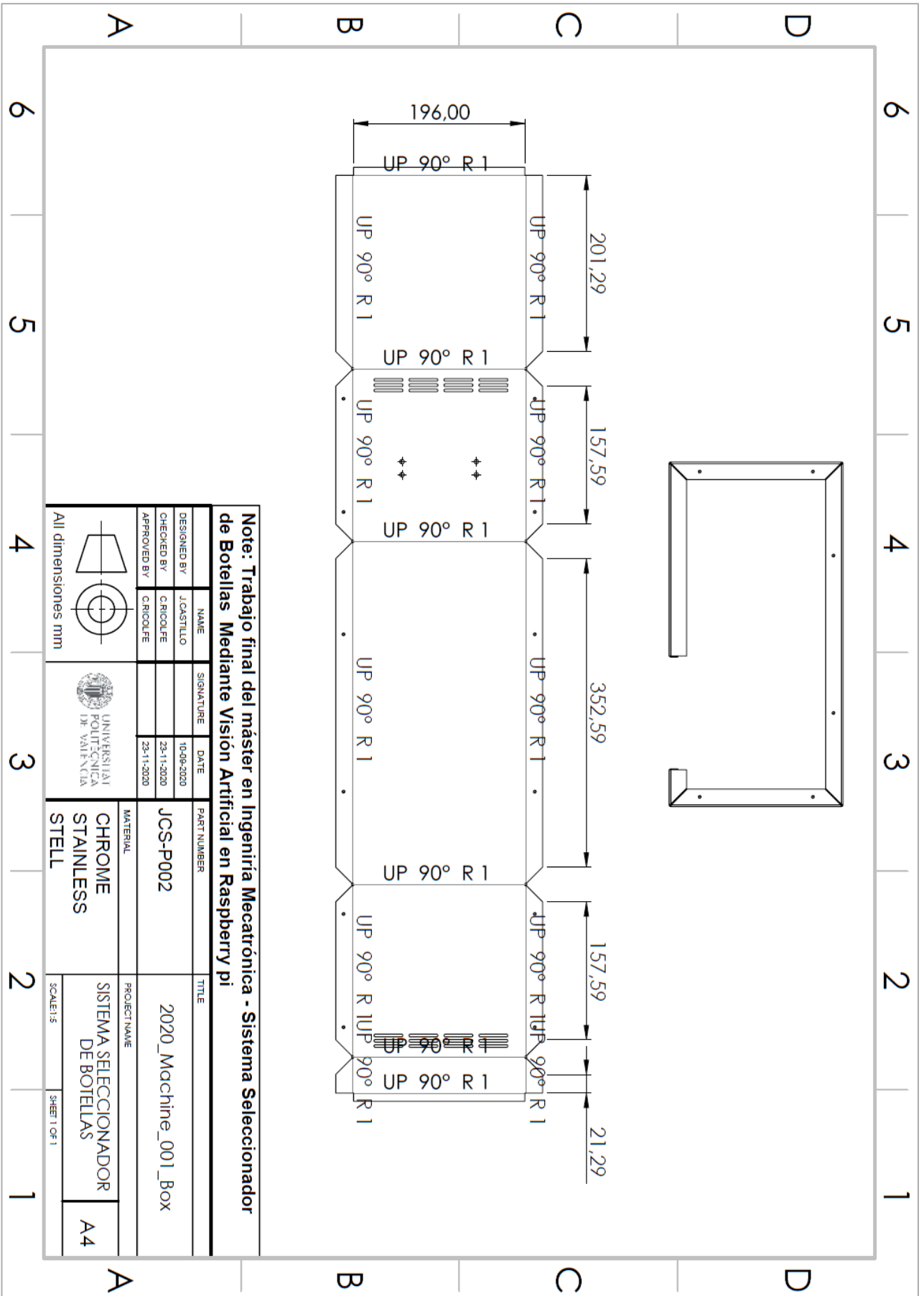


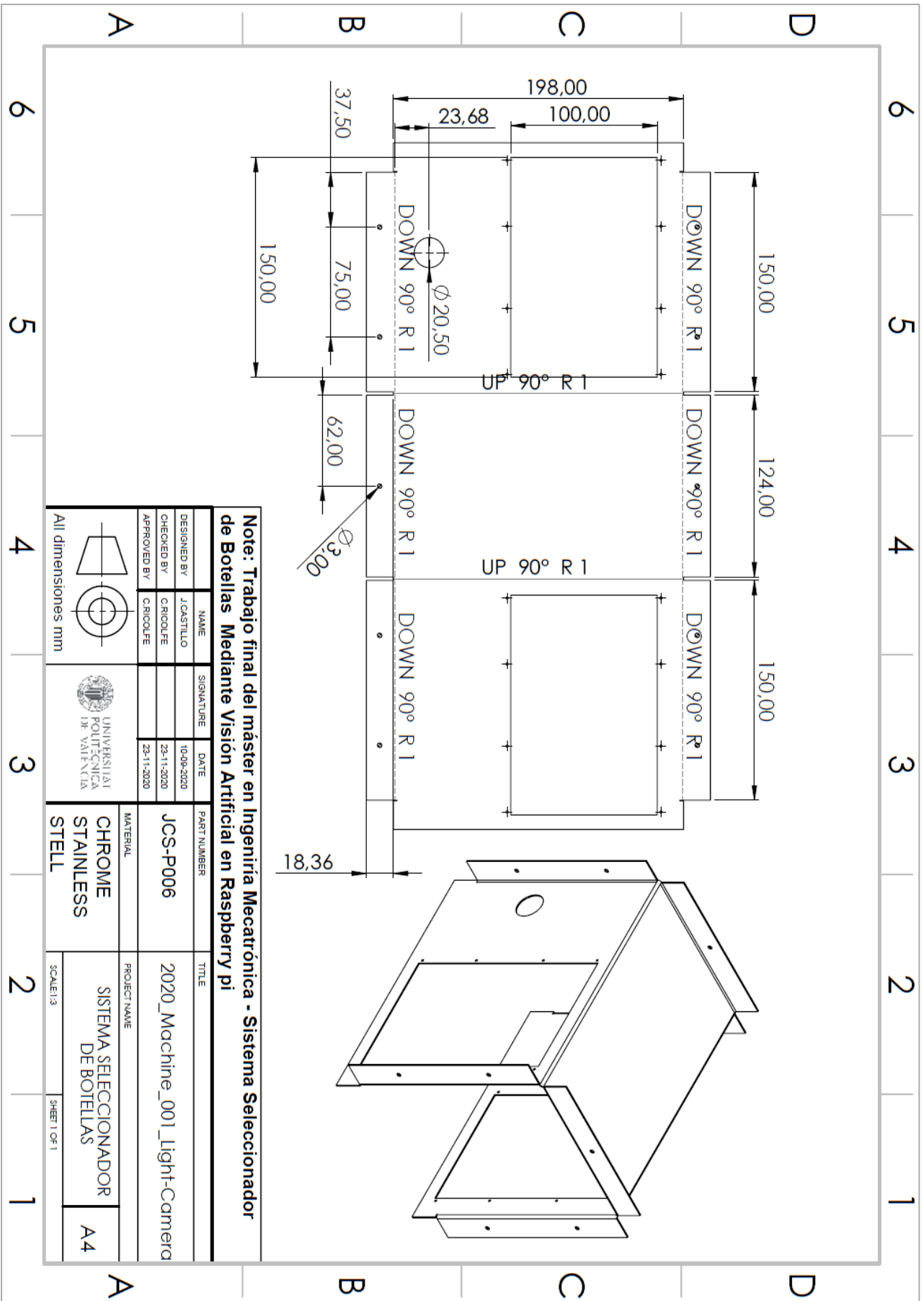
## Anexo 2 - Planos

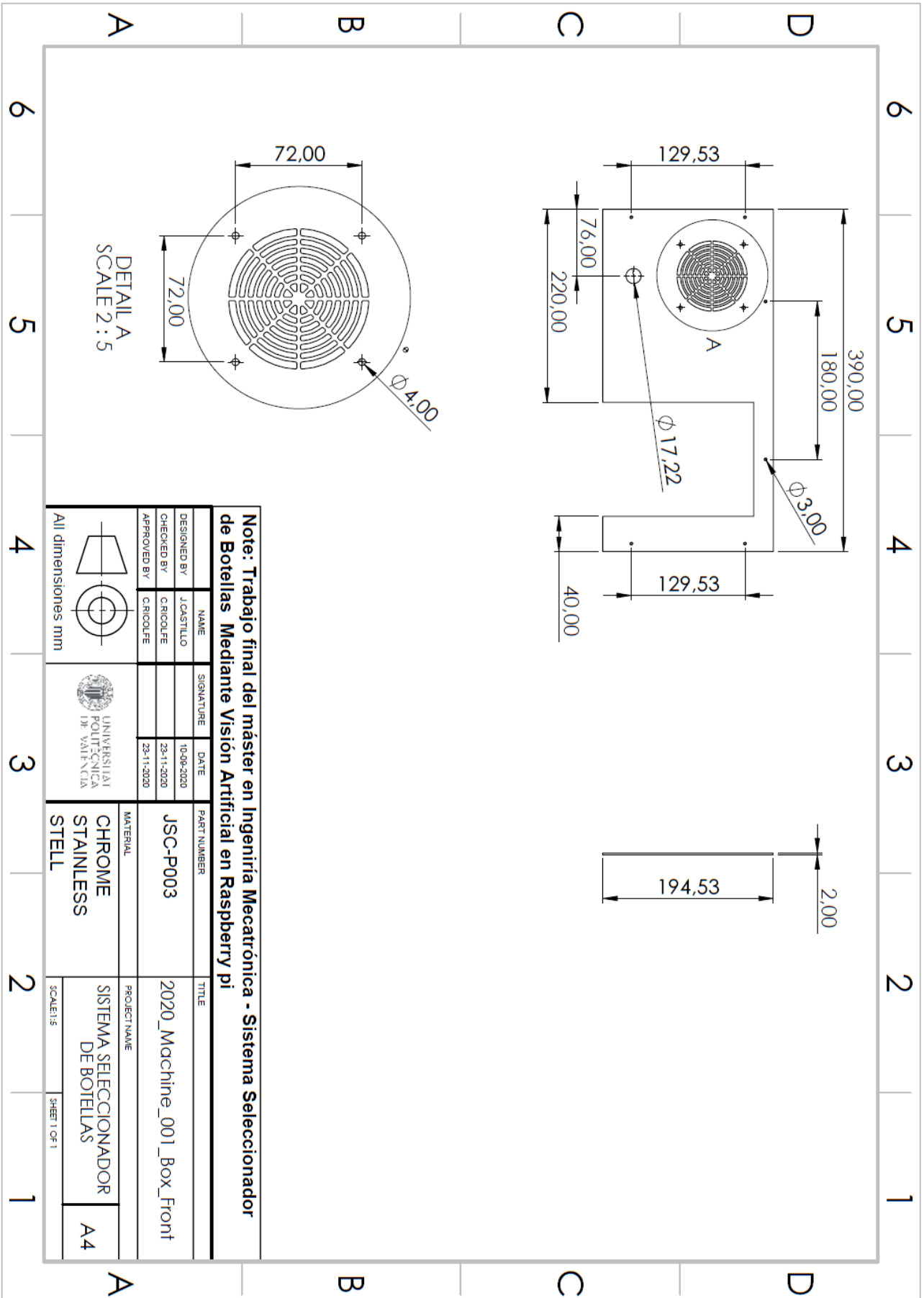


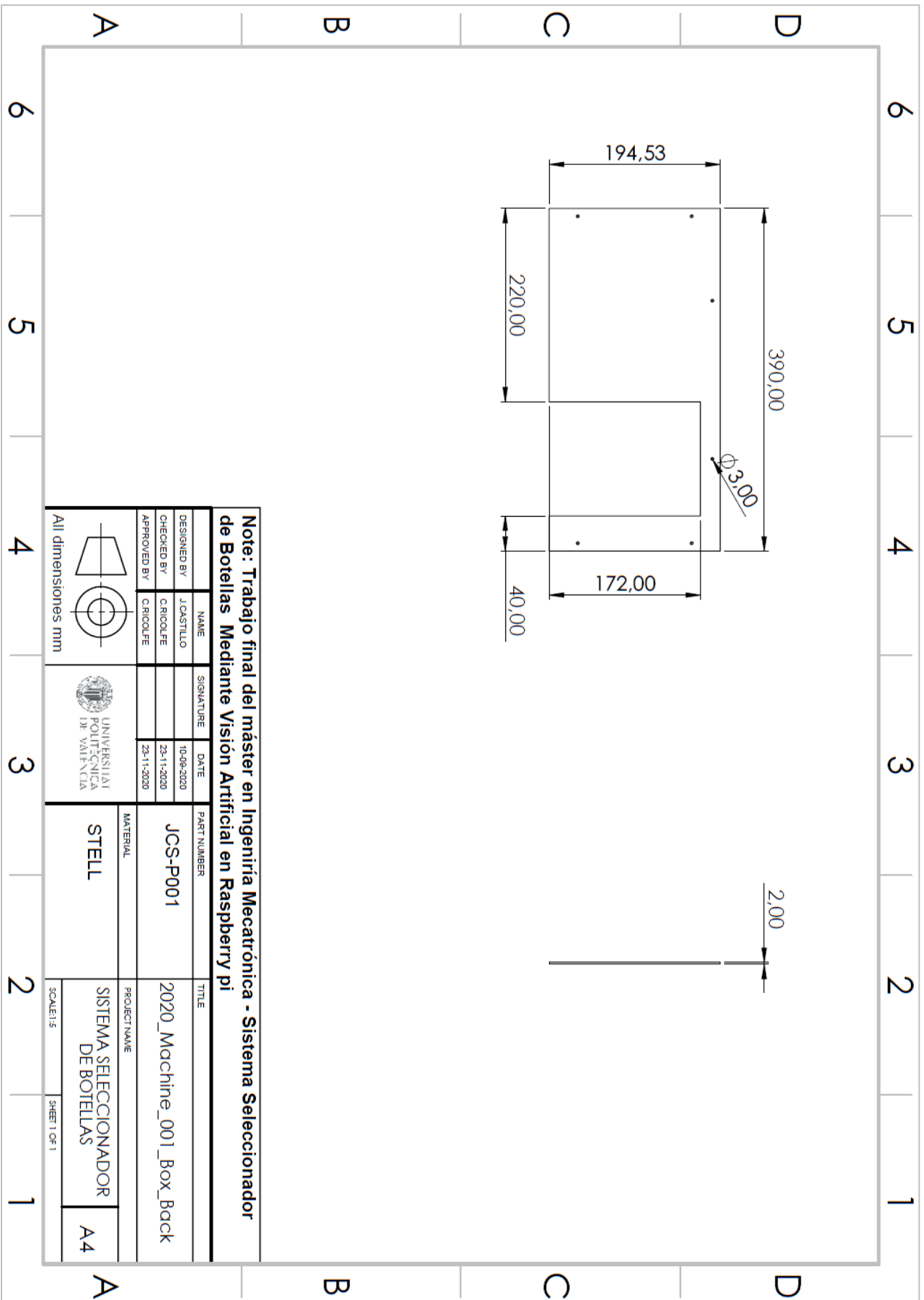




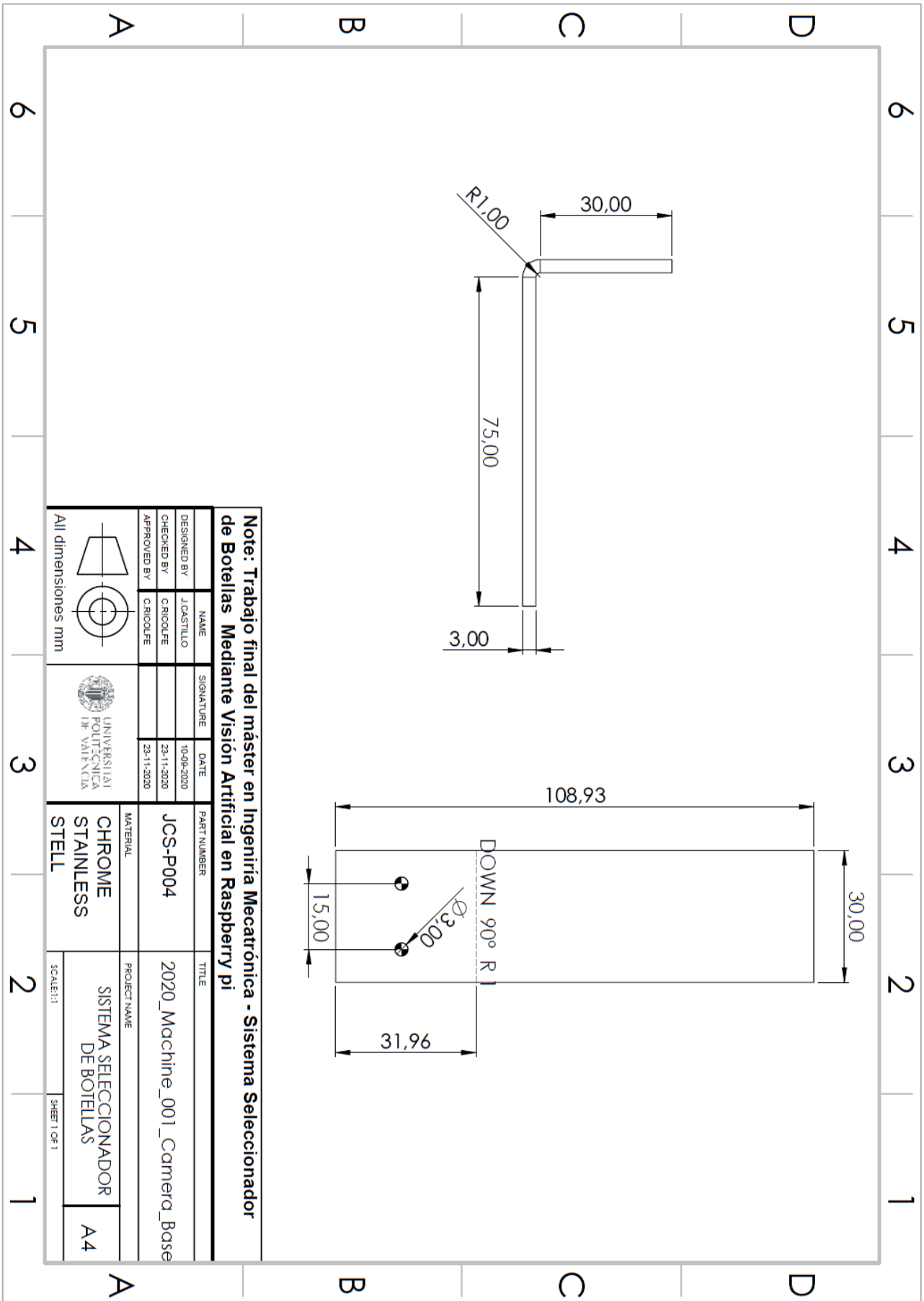






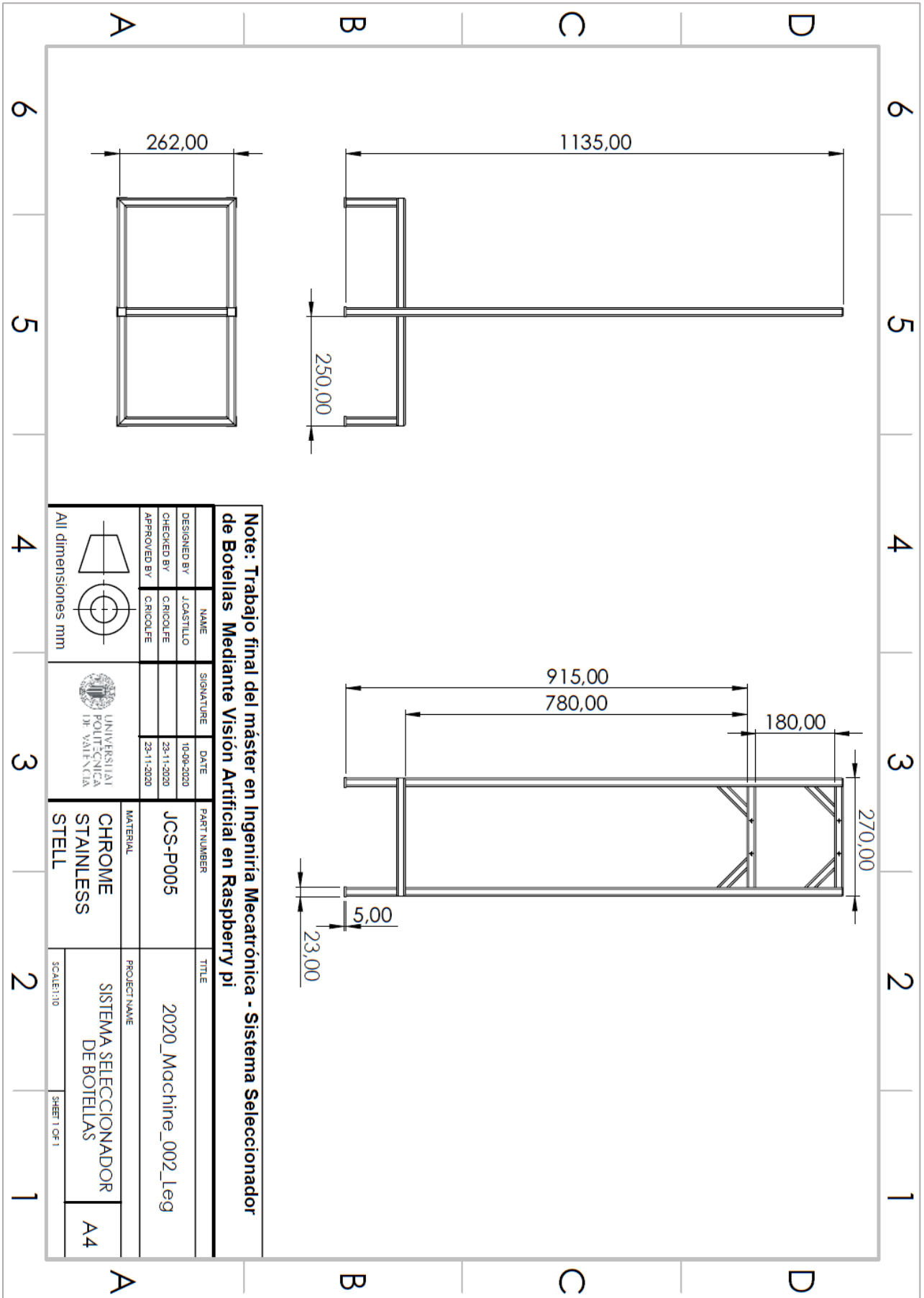




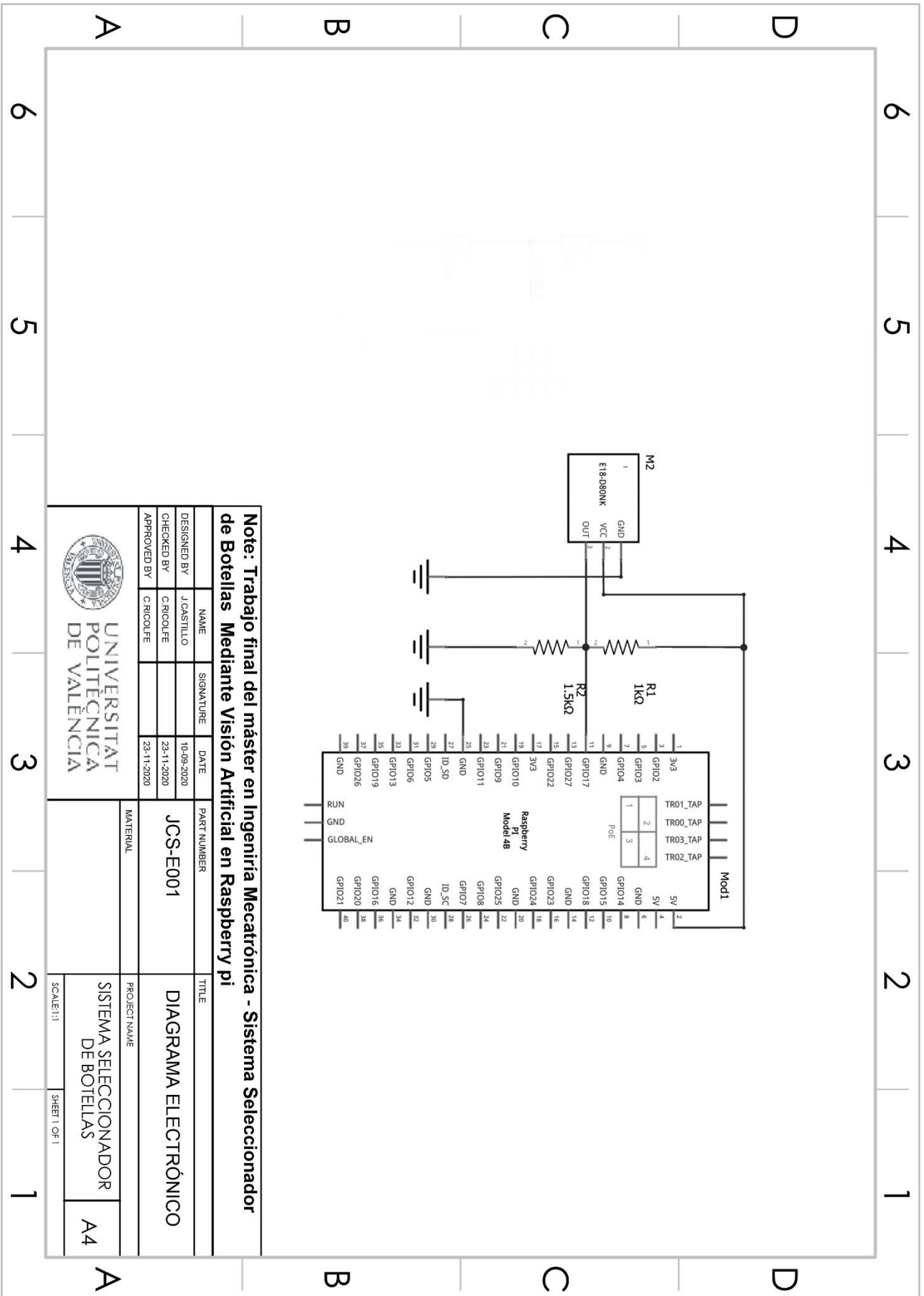


**Note: Trabajo final del máster en Ingeniería Mecatrónica - Sistema Seleccionador de Botellas Mediante Visión Artificial en Raspberry pi**

DESIGNED BY	NAME	SIGNATURE	DATE	PART NUMBER	TITLE
JCASTILLO	JCASTILLO		10-06-2020	JCS-P004	2020_Machine_001_Camera_Base
C.RICOLFE	C.RICOLFE		23-11-2020		
C.RICOLFE	C.RICOLFE		23-11-2020		
MATERIAL		PROJECT NAME		SCALE: 1:1	
CHROME STAINLESS STEEL		SISTEMA SELECCIONADOR DE BOTELLAS		SHEET 1 OF 1	
All dimensiones mm		UNIVERSITAT POLITÈCNICA DE VALÈNCIA		A4	



### Anexo 3 - Esquema eléctrico



## Anexo 4 - Cotizaciones y justificantes de precios

28/11/2020

Gmail - Your invoice for eBay purchases: BASLER ACA640-100GM CAMERA WFT CENTERING (301936640351#)



Jhulius Castillo Suárez <jhulius0691@gmail.com>

### Your invoice for eBay purchases: BASLER ACA640-100GM CAMERA WFT CENTERING (301936640351#)

1 mensaje

eBay <ebay@ebay.com>  
Para: jhulius0691@gmail.com

25 de agosto de 2020, 15:39



eBay sent this message to jhulius castillo (jhulius1511).

Your registered name is included to show this message originated from eBay. [Learn more.](#)

#### Invoice

Dear jhulius1511,

**Pay Now**

Thank you for shopping on eBay! Your total amount due is €334,34. More details about your purchase are included below.

Item #	Item Title	Quantity	Price	Amount
301936640351	<a href="#">BASLER ACA640-100GM CAMERA WFT CENTERING</a>	1	€334,34	€334,34

Subtotal: €334,34

Shipping and handling via USPS Priority Mail: €39,15

**Total: €373,49**

Email reference id: [#daba582156404ddb930939fff9c8d61e#]

[Learn More](#) to protect yourself from spoof (fake) emails.

eBay sent this email to you at [jhulius0691@gmail.com](mailto:jhulius0691@gmail.com) about your account registered on [www.ebay.com](http://www.ebay.com).

eBay will periodically send you required emails about the site and your transactions. Visit our [Privacy Notice](#) and [User Agreement](#) if you have any questions.

Copyright © 2016 eBay Inc. All Rights Reserved. Designated trademarks and brands are the property of their respective owners. eBay and the eBay logo are trademarks of eBay Inc. eBay Inc. is located at 2145 Hamilton Avenue, San Jose, CA 95125.

<https://mail.google.com/mail/u/0?ik=1c85cee728&view=pt&search=all&permthid=thread-f%3A1543642644524961548&simpl=msg-f%3A1543642...> 1/1

29/11/2020

11/28/20 14:32: A Cart has been forwarded to you from the Thorlabs website

**Data:** 28/11/2020 [20:33:37 CET]  
**Da:** webmaster@thorlabs.com  
**A:** jhucassu@etsid.upv.es  
**Cc:** jhulius0691@gmail.com  
**Oggetto:** 11/28/20 14:32: A Cart has been forwarded to you from the Thorlabs website

To receive future email confirmations in text only format, please go to the "Update your Billing Address" section of your "My Account" page at <https://www.thorlabs.com>

Jhulius Castillo Suárez,

jhulius0691@gmail.com has forwarded a cart from the [Thorlabs](https://www.thorlabs.com) website.

To retrieve the contents of this cart, go to the [Rapid Order](#) page and enter the code shown below into the "Retrieve Cart" field:

**Retrieval Code: djt5huGQnuMW**

Below is a listing of the products in this saved cart:

**Product List - 11/28/20 14:32**

Part Number	Description	Qty	Price
<a href="#">MVL5WA</a>	4.5 mm EFL, f/1.4, for 1/2" C-Mount Format Cameras, with Lock	1	173,13 €
		Total Costs	173,13 €


**\*\* NOTE: \*\***

- This cart will be saved for 14 Days from today.
- An order must be placed before the cart expires.
- Prices are subject to change without notice.

**\*\*\*\* This is an automatically generated email. \*\*\*\***

[https://webmail.upv.es/imp/view.php?view\\_token=VewnmI37sZ2SLc5H-ABFBji&actionID=print\\_attach&buid=1555&id=1&mailbox=SU5CT1g&toke...](https://webmail.upv.es/imp/view.php?view_token=VewnmI37sZ2SLc5H-ABFBji&actionID=print_attach&buid=1555&id=1&mailbox=SU5CT1g&toke...) 1/1

## Shopping cart

Remove	VS ID	Image	Product(s)	Price	Qty.	Total
<input type="checkbox"/>	99118		DCM Sistemas BKL1510A-W00C	€983.99 excl tax	<input type="text" value="1"/>	€983.99 excl tax



### Discount Code

Enter your coupon here

### Estimate International Shipping

Enter your destination to get a shipping estimate

Country:  \*

State/Province/ Country:

Zip/Postcode:

Sub-Total: €983.99 excl tax

Shipping: Calculated during checkout

VAT/Sale Tax/TVA 0%: €0.00

**Total:** Calculated during checkout

### Standard Shipping (€0.00 excl tax)

Ship using our standard carrier for up to 1kg total weight.



## Elige las opciones de envío

Continuar

### Envío 1 de 1

#### Envío de Amazon.es [\(Más información\)](#)

Enviar a: Jhulius Castillo Suárez, Avenida Cardenal Benlloch 71  
Piso 3, puerta 9, Valencia, Comunidad Valencia, 46021 España

- AABCOOLING Black Silent Fan 8 - Un Silencioso y Muy Efectivo Ventilador PC, Fan 80mm, Ventilador 12V, Ventilador Laptop, Fan Cooler, 50 m3/h, 2000 RPM 15,6 dB (A)**  
6,99 € - Cantidad: 1  
Vendido por: AAB Seller ES
- SanDisk Ultra Tarjeta de memoria microSDHC con adaptador SD, hasta 98 MB/s, rendimiento de apps A1, Clase 10, U1, 32 GB**  
5,90 € - Cantidad: 1  
Vendido por: Amazon EU S.a.r.L.
- NanoCable 10.20.0401 - Cable de red Ethernet RJ45 Cat.6 UTP AWG24, 100% cobre, Gris, latigullo de 1mts**  
1,75 € - Cantidad: 4  
Vendido por: Amazon EU S.a.r.L.
- Raspberry Pi Spain RAS-4-4G - Placa Base Pi 4 Modelo B / 4 GB SDRAM (1822096) - Raspberry-Pi**  
65,00 € - Cantidad: 1  
[Ver el resumen del pedido para ver el precio después del descuento](#)  
Ordenadores personales - Nuevo  
Vendido por: S & P Deals
- TP-Link TL-PoE150S Gigabit Ethernet PoE Inyector, Plug-and-Play, 2 puertos 10/100/1000Mbps RJ45, 1 puerto de 48VDC, 15,4 W (máximo 48 V DC) - Tp-Link**  
18,36 € - Cantidad: 1  
Ordenadores personales - Nuevo  
Vendido por: Amazon EU S.a.r.L.
- Velleman K/MOWM - Cable de señal (60 m, AWG24)**  
13,90 € - Cantidad: 1  
Vendido por: Amazon EU S.a.r.L.
- E18-D80NK Interruptor de sensor de infrarrojos para evitar Interruptor de sensor de detección de obstáculos de infrarrojos ajustable**  
11,89 € - Cantidad: 1  
Vendido por: Partgry

[Modificar cantidades o eliminar](#)

### Elige una opción de envío

#### ¿Por qué pagar por envíos rápidos?


Jhulius Castillo, para obtener envío rápido y GRATIS, es "GRATIS Envío exprés con el periodo de prueba gratis de Prime."

- GRATIS Envío exprés con el periodo de prueba gratis de **amazon prime** — recibelo entre el miércoles, 2 de dic y el jueves, 3 de dic
- GRATIS Envío económico — recibelo viernes, 4 de dic
- 3,99 € Envío estándar — recibelo jueves, 3 de dic
- 4,99 € Envío exprés — recibelo entre el miércoles, 2 de dic y el jueves, 3 de dic
- Entrega programada — dom., 6 dic., 11:00 - 13:00  
[Cambiar fecha y horario de entrega](#)

Continuar

¿Tienes un cheque regalo o un código promocional? Te pediremos que lo introduzcas en el momento del pago.

[Condiciones de uso](#) | [Aviso de privacidad](#) © 2011-2020, Amazon.com, Inc.



Home  
Tienda Online SITASA
Tienda Online  
Venta de productos
Contacto  
y soporte técnico
Zona Clientes  
login / registro
Pedidos  
Seguimiento

2 artículos en total: 281,11 €
IVA incluido

Cesta
Datos Facturación
Tipo de Envío
Formas de Pago
Realizar el Pago

Descripción	Precio	%Dto	U. Venta	Cant.	Total
INOX 20-20-1 TUBO AISI-304	8,6761 €	0,00 %	DMT	4,20	36,44 €
Comentario: <input type="text" value="Inserte un comentario para el producto"/> <input type="button" value="Guardar comentario"/>					
INOX 2000-1000-2 PLANCHA A-304	7,6450 €	0,00 %	0/KG	32,00	244,67 €
Comentario: <input type="text" value="Inserte un comentario para el producto"/> <input type="button" value="Guardar comentario"/>					

Resumen de Impuestos

	Base	% IVA	Iva
	232,32 €	21,00 %	48,79 €

Observaciones del pedido:

**Totales del pedido**

SubTotal:	281,11 €
Portes:	0,00 €
Gastos F.Pago:	0,00 €
<b>TOTAL:</b>	<b>281,11 €</b>

**PEDIDO MÍNIMO DE 50 € IVA INCLUIDO**

[← Seguir comprando](#)
[Siguiente →](#)


SITASA SUMINISTROS INDUSTRIALES S.A. | Jarama 52, Pol. Industrial | 45006 | Toledo | Toledo | Tel. 925 23 22 00 | [sitasa@sitasa.com](mailto:sitasa@sitasa.com)

Suministro de materiales, equipos y repuestos para el mantenimiento industrial, con alta diversificación dentro de los distintos sectores productivos, ofreciendo un servicio integral de soluciones industriales de gran agilidad y capacidad operativa. Comercialización a escala nacional de más de 200.000 referencias. Asesoramiento técnico y desarrollo de formación a medida. Gestión externa de aprovisionamientos, logística y stock.

[www.sitasa.com](http://www.sitasa.com)


[comercio electrónico - mdsai.com](http://comercio.electronico-mdsai.com)


[Aviso Legal](#) [Política de privacidad](#) [Política de Cookies](#)



GOBIERNO  
DE ESPAÑA

MINISTERIO  
DE INDUSTRIA, ENERGÍA  
Y TURISMO





UNIÓN EUROPEA

PROYECTO COFINANCIADO  
POR EL FONDO EUROPEO DE  
DESARROLLO REGIONAL  
(FEDER)

Una manera de hacer Europa



## Anexo 5 – Especificaciones de dispositivos

Especificaciones técnicas Raspberry pi 4B.

### Specification

<b>Processor:</b>	Broadcom BCM2711, quad-core Cortex-A72 (ARM v8) 64-bit SoC @ 1.5GHz
<b>Memory:</b>	2GB, 4GB or 8GB LPDDR4 (depending on model)
<b>Connectivity:</b>	2.4 GHz and 5.0 GHz IEEE 802.11b/g/n/ac wireless LAN, Bluetooth 5.0, BLE Gigabit Ethernet 2 × USB 3.0 ports 2 × USB 2.0 ports.
<b>GPIO:</b>	Standard 40-pin GPIO header (fully backwards-compatible with previous boards)
<b>Video &amp; sound:</b>	2 × micro HDMI ports (up to 4Kp60 supported) 2-lane MIPI DSI display port 2-lane MIPI CSI camera port 4-pole stereo audio and composite video port
<b>Multimedia:</b>	H.265 (4Kp60 decode); H.264 (1080p60 decode, 1080p30 encode); OpenGL ES, 3.0 graphics
<b>SD card support:</b>	Micro SD card slot for loading operating system and data storage
<b>Input power:</b>	5V DC via USB-C connector (minimum 3A <sup>1</sup> ) 5V DC via GPIO header (minimum 3A <sup>1</sup> ) Power over Ethernet (PoE)-enabled (requires separate PoE HAT)
<b>Environment:</b>	Operating temperature 0–50°C
<b>Compliance:</b>	For a full list of local and regional product approvals, please visit <a href="https://www.raspberrypi.org/documentation/hardware/raspberrypi/conformity.md">https://www.raspberrypi.org/documentation/hardware/raspberrypi/conformity.md</a>
<b>Production lifetime:</b>	The Raspberry Pi 4 Model B will remain in production until at least January 2026.

Para más información visitar el enlace:

<https://static.raspberrypi.org/files/product-briefs/200521+Raspberry+Pi+4+Product+Brief.pdf>

## Especificaciones técnicas cámara Basler acA640-100gm/gc.

### TECHNICAL DETAILS

### Specifications

Basler <i>ace</i>	Resolution (H x V pixels)	Sensor	Sensor Technology	Sensor Size (optical)	Pixel Size (µm)	Frame Rate	Power Consumption (PoE/AUX)	Weight (typical)
acA640-90gm/gc	659 x 494	Sony ICX424	Progressive Scan CCD	1/3"	7.4 x 7.4	90	3.1 W/2.7 W	<90g
acA640-100gm/gc	659 x 494	Sony ICX618	Progressive Scan CCD	1/4"	5.6 x 5.6	100	2.3 W/2.0 W	<90 g
acA645-100gm/gc	659 x 494	Sony ICX414	Progressive Scan CCD	1/2"	9.9 x 9.9	100	3.6 W/3.3 W	<90g
acA750-30gm/gc	752 x 580	Sony ICX409	Interlaced Scan CCD	1/3"	6.5 x 6.25	30	2.5 W/2.3 W	<90 g
acA780-75gm/gc	782 x 582	Sony ICX415	Progressive Scan CCD	1/2"	8.3 x 8.3	75	3.6 W/3.3 W	<90g
acA1300-30gm/gc	1296 x 966	Sony ICX445	Progressive Scan CCD	1/3"	3.75 x 3.75	30	2.5 W/2.2 W	<90 g
acA1600-20gm/gc	1628 x 1236	Sony ICX274	Progressive Scan CCD	1/1.8"	4.4 x 4.4	20	3.4 W/2.9 W	<90 g
acA2000-50gm/gc*	2048 x 1088	CMOSIS CMV2000	CMOS, global shutter	2/3"	5.5 x 5.5	50		<90 g
acA2040-25gm/gc*	2048 x 2048	CMOSIS CMV4000	CMOS, global shutter	1"	5.5 x 5.5	25		<90 g
acA2500-14gm/gc	2592 x 1944	Aptina MT9P	CMOS, rolling shutter	1/2.5"	2.2 x 2.2	14	2.5 W/2.2 W	<90 g

\*Available Q2/2012

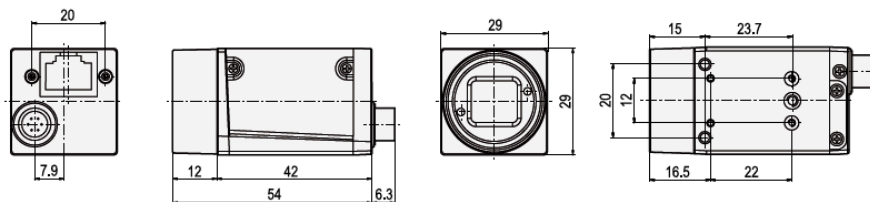
### Specifications Applicable For All ace GigE Models:

Mono / Color	Mono / Color
Interface	Fast Ethernet (100 Mbit/s) or Gigabit Ethernet (1000 Mbit/s)
Video Output Format	Mono 8, Mono 12, Mono 12 Packed, YUV 4:2:2 Packed, YUV 4:2:2 (YUYV) Packed, Bayer BG 8, Bayer BG 12, Bayer BG 12 Packed, acA750-30gc: Mono 8, YUV 4:2:2 Packed, YUV 4:2:2 (YUYV) Packed only
Synchronization	Via external trigger, via the Ethernet connection or free run
Exposure Control	Via external trigger or programmable via the camera API
Housing Size (L x W x H)	42 mm x 29 mm x 29 mm
Housing Temperature	Up to 50°C
Lens Mount	C-mount, CS-mount (except acA2000-50gm/gc and acA2040-25gm/gc)
Digital I/O	1 opto-isolated input / 1 opto-isolated output
Power Requirements	Via Power over Ethernet (802.3af) or + 12VDC (±10%) via the camera's 6-pin Hirose connector
Conformity	CE, FCC, IP30, RoHS, PoE (802.3af), UL (in preparation), GigE Vision, GenICam
Driver	Basler pylon SDK including filter and performance driver
Operating System	Windows, Linux - 32 bit and 64 bit

Specifications are subject to change without prior notice.

For detailed technical information, please see the camera manual that can be found on our website: [www.baslerweb.com/manuals](http://www.baslerweb.com/manuals)

### Dimensions (in mm)



## Especificaciones técnicas del adaptador PoE TL-POE150S.

CARACTERÍSTICAS DE HARDWARE	
Interfaz	2 puertos 10/100/1000Mbps RJ45 con AUTO Negociación/AUTO MDI/MDIX 1 puerto de 48VDC
Medios de Red	10BASE-T: cable UTP categoría 3, 4, 5 (100 metros máximo) EIA/TIA-568 100Ω STP (máximo 100 m) 100BASE-TX: cable UTP categorías 5, 5e (máximo 100 metros) EIA/TIA-568 100Ω STP (máximo 100 m) 1000BASE-T: cable UTP categoría 5, 5e o 6 (100 metros máximo)
Fuente de Alimentación	15,4 W (máximo 48 V DC)
Indicador de LED	PWR (alimentación)
Dimensiones	3.2*2.1*0.9 pulgadas(80.8*54*24 mm)
CARACTERÍSTICAS DE SOFTWARE	
Funciones Básicas	Soporta dispositivos compatibles con el estándar IEEE 802.3af Suministra alimentación eléctrica hasta 100 metros Identifica automáticamente los requisitos de alimentación necesarios
OTROS	
Certificaciones	FCC, CE
Contenido del Paquete	Gigabit PoE Injector TL-POE150S Power Adapter RJ45 Cable Installation Guide
Requisitos del sistema	Microsoft® Windows® 98SE, NT, 2000, XP, Vista™ o Windows 7, MAC® OS, NetWare®, UNIX® o Linux.
Factores Ambientales	Temperatura de funcionamiento: 0°C ~ 40°C (32°F ~ 104°F) Temperatura de almacenamiento: -40°C ~ 70°C (-40°F ~ 158°F) Humedad: 10% ~ 90% sin condensación Humedad de almacenamiento: 5% ~ 90% sin condensación