

Diseño y construcción de un equipo de medición dimensional para piezas del sector agrícola para la implementación en la empresa Aceros Markert S.A.S.

Autor:  
Kevin Markert López

UNIVERSIDAD AUTÓNOMA DE BUCARAMANGA  
FACULTAD DE INGENIERÍAS FISICOMECÁNICAS  
INGENIERÍA MECATRÓNICA  
2020

Diseño y construcción de un equipo de medición dimensional para piezas del sector agrícola para la implementación en la empresa Aceros Markert S.A.S.

Autor:  
Kevin Markert López

Trabajo de grado para optar por el título de Ingeniero Mecatrónico

Director:  
Sergio Andrés Ardila Gómez

UNIVERSIDAD AUTÓNOMA DE BUCARAMANGA  
FACULTAD DE INGENIERÍAS FISICOMECAÑICAS  
INGENIERÍA MECATRÓNICA

2020

Nota De Aprobación

Trabajo de grado titulado” Diseño y construcción de un equipo de medición dimensional para piezas del sector agrícola para la implementación en la empresa Aceros Markert S.A.S.”, presentado por el estudiante Kevin Markert Lopez para optar por el título de Ingeniero Mecatrónico



---

Firma jurado



---

Firma jurado



Aprobado  
Sergio Ardila

---

Firma director

## Tabla de contenido

1. Introducción.....	8
2.Objetivos .....	9
2.1 Objetivo general.....	9
2.2 Objetivos específicos .....	9
3. Estado del arte .....	10
4.Marco teórico.....	13
4.1 Proceso de manufactura:.....	13
4.1.1 Metales ferrosos:.....	14
4.1.2 Forja: .....	14
4.1.3 Esmerilado: .....	16
4.1.4 Granallar: .....	17
4.1.5 Perforado: .....	17
4.1.6 Biselado: .....	18
4.1.7 Ranurado: .....	18
4.1.8 Tratamiento térmico: .....	18
4.1.9 Grateado: .....	19
4.2 Sistema de unidades .....	20
4.3 Filtro Gaussiano.....	20
4.4 Filtro Canny: .....	21
4.5 Sistema de visión artificial.....	23
6. Desarrollo del proyecto.....	27
6.1 Levantamiento de los requerimientos del equipo.....	27
6.2 Parametrización .....	28
6.3 Selección .....	29
6.3.1 Entornos de desarrollo .....	30
6.3.2 Hardware.....	31
6.3.3 Cámara de visión .....	32
6.3.4 Arranque del motor.....	33
6.3.5 Sistema de clasificación .....	34
6.4.1 Cálculos. ....	35

6.4.2 Diseño Banda .....	41
6.5 Circuitos.....	43
6.5.1 Circuito potencia.....	43
6.5.2 Circuito de control. ....	44
6.6 Visión artificial.....	44
6.6.1 Conexión Raspberry.....	44
6.6.2 Código visión. ....	45
7 Equipo construido.....	47
7.1 Construcción parte mecánica. ....	47
7.2 Conexión subsistemas.....	48
8 Pruebas y Resultados .....	50
8.1 Largo de pieza: .....	51
8.2 Ancho de pieza: .....	51
8.3 Precisión: .....	52
8.3.1 Aciertos largo de pieza:.....	52
8.3.2 Aciertos ancho de pieza: .....	52
9 Conclusiones.....	53
10 Bibliografía .....	54
11. Anexos .....	56
11.1 Anexo A .....	56
11.2 Anexo B. ....	58
11.3 Anexo C.....	67

## Lista de ilustraciones

Ilustración 1 Forja de herrero.[10]	15
Ilustración 2 Forja con martinete.[10]	15
Ilustración 3 Piedra de esmeril.[11]	16
Ilustración 4 Piezas granalladas. [Autor]	17
Ilustración 5 Cepillo de grateado.[17]	19
Ilustración 6 Sistema visión artificial.[21]	25
Ilustración 7 Metodología en V.[Autor]	26
Ilustración 8 Bosquejo del sistema.[Autor]	27
Ilustración 9 Guarda. [Autor]	28
Ilustración 10 Planos guarda.[Autor]	29
Ilustración 11 Visual code.[Autor]	30
Ilustración 12 Raspberry.[Autor]	31
Ilustración 13 Cámara Logitech C525.[Autor]	32
Ilustración 14 Variador.[Autor]	33
Ilustración 15 Motor lineal.[Autor]	34
Ilustración 16 Coeficientes de fricción. [23]	36
Ilustración 17 Coeficiente de fricción [23]	37
Ilustración 18 Factor de servicio. [23]	37
Ilustración 19 Factor de temperatura[23]	38
Ilustración 20 Par de torsión.[23]	40
Ilustración 21 Prototipo.[Autor]	41
Ilustración 22 Sistema clasificado.[Autor]	42
Ilustración 23 Circuito de potencia.[Autor]	43
Ilustración 24 Circuito de control.[Autor]	44
Ilustración 25 Diagrama de flujo.[Autor]	45
Ilustración 26 Construcción parte mecánica.[Autor]	47
Ilustración 27 Conexión subsistemas.[Autor]	48
Ilustración 28 Prototipo.[Autor]	49
Ilustración 29 Pruebas prototipo.[Autor]	50

## Índice de tablas

Tabla 1 Matriz QFD entorno de desarrollo.[Autor].....	30
Tabla 2 Matriz QFD hardware[Autor].....	31
Tabla 3 Matriz QFD cámara de visión.....	32
Tabla 4 Matriz QFD arranque motor.....	33
Tabla 5 Matriz QFD sistema de clasificad.....	34
Tabla 6 Resultados largo de pieza.[Autor] .....	51
Tabla 7Resultados ancho de pieza.[Autor].....	51
Tabla 8 Precisión largo.[Autor] .....	52
Tabla 9 Precisión ancho de pieza.[Autor].....	52

## 1. Introducción

Los sistemas de calidad en la industria metal-mecánica tienden a ser procesos tediosos y un poco lentos respecto a la velocidad de demanda de los productos, la gran mayoría de estos procesos se realizan de manera manual; para la realización de esto se requiere un operario realizando esta gestión de calidad. En este proyecto se realizó un equipo que permite generar un sistema de calidad para verificación de medidas de forma automatizada.

Un sistema de calidad automatizado permite la verificación de una mayor cantidad de piezas verificadas que cumplan con las medidas estandarizadas, lo cual también permite validar el estado de las piezas como correctas o incorrectas

El avance que ha tenido la visión artificial en los últimos años permite usar diferentes metodologías para el desarrollo de programas con los cuales se pueden obtener las medidas de las piezas para lograr comparar éstas con las medidas estandarizadas por el fabricante. El uso de estas nuevas metodologías permite facilitar la programación y automatización en las tomas de datos.

El desarrollo del proyecto se realizó por medio de una investigación del estado del arte, para obtener una base sobre los conocimientos en los sistemas de calidad y visión artificial, posteriormente se realizó una selección de los actuadores y metodologías para el desarrollo del proyecto. Finalmente, se pasó a realizar la construcción y programación del sistema automatizado teniendo en cuenta el resto de activaciones que debe tener el programa.

El proyecto cuenta con una banda transportadora en la cual las piezas avanzan por la primera etapa donde se realiza la toma de la imagen y se procesa esta, la segunda estación es el sistema de clasificación, después de codificar la información se toma la decisión si la pieza esta apta o no para continuar con el proceso.

Se hace constar que, el propósito general del proyecto es la realización de un sistema de medición dimensional para piezas de maquinaria agrícola, para la empresa Aceros Markert, con el fin de aumentar la eficiencia del sistema de calidad en la empresa.

## 2.Objetivos

### 2.1 Objetivo general

- Desarrollar un equipo de medición dimensional para piezas del sector agrícola utilizando técnicas de visión artificial para la implementación en la empresa Aceros Markert S.A.S.

### 2.2 Objetivos específicos

- Caracterizar y parametrizar la pieza de maquinaria agrícola a desarrollar por la empresa Aceros Markert S.A.S.
- Diseñar el sistema electrónico de control y visión artificial para el equipo de medición dimensional.
- Seleccionar materiales, actuadores mecánicos, electromecánicos y dispositivos electrónicos para la construcción del equipo de medición dimensional para piezas de maquinaria agrícola.
- Construir el equipo de medición dimensional para piezas de maquinaria agrícola.
- Realizar pruebas de funcionamiento del equipo implementado en la empresa Aceros Markert S.A.S.

### 3. Estado del arte

La visión artificial es un tema de estudio que ha avanzado a pasos agigantados en los últimos años, pero ésta no solo se queda a nivel académico, son diversas las aplicaciones que se pueden encontrar en la industria hoy en día; lo cual facilita la presente investigación. Unas de las aplicaciones en la industria pueden ser para la sección de calidad o toma de medidas de piezas. Primeramente, se cita una patente desarrollada en West Palm Beach, florida, Estados Unidos, La patente llamada ARTIFICIAL VISION INSPECTION METHOD AND SYSTEM en el año de 2007 [2]. En esta patente se muestra como por medio de los filtros: enhancement filter, grey scale, threshold filter, unrelated structure filter, verificaban el contorno de las piezas y clasificaban si las piezas cumplían con la geometría programada, con esto buscaban encontrar anomalías en las piezas realizadas manualmente.

En el año 2012, en la revista de la unión europea FSE [3], En esta revista de la unión europea desarrollada por la secretaria de estado de educación y formación profesional, se muestran las diversas aplicaciones de la visión artificial en la industria, explica un poco los procesos de cómo deben ser montados los procesos de visión artificial y los componentes que requiere cada sistema, y las cosas que se deben tener en cuenta para que el sistema tenga una mejor toma de imagen; se deben tener en cuenta: el tipo de cámara, foco, tarjeta de adquisición y computador. También dice que las empresas deben empezar a usar todas estas nuevas tecnologías de visión artificial para la verificación de procesos manufactureros.

En CLASIFICATION SYSTEM BASED ON COMPUTER VISION de Porras y Cruz de 2014 [4] realizado en el año 2014, lograron la clasificación de objetos por medio de su color, el sistema descrito consiste en una banda transportadora la cual en el medio cuenta con una cámara para visión, la cual es la encargada de controlar el sistema por medio de Matlab, al final de la banda transportadora hay tres recipientes en los cuales empacaban las piezas clasificadas por sus colores por medio de motores paso a paso. Por medio de este sistema obtuvieron una autonomía y precisión del 100% en la clasificación de las piezas.

En el proyecto Development of contactless integrated interface of complex production lines de Ostroukh, A., Morozova, T., Nikonov, V.[5] en 2014 muestra diversas metodologías y desarrollos para el control de las líneas de producción. Ellos se enfocaron en la visión y cómo se deben programar muchas metodologías de visión para que sea lo más eficiente. Los autores compararon todas las metodologías realizadas entre ellos. Las pruebas las realizaron tomando videos a los ojos de las personas. Lograron realizar el control por medio de un sistema fuzzy, el cual lo implementaron en varios softwares como Matlab y C++.

En el proyecto de grado Inspección de calidad para un sistema de producción industrial basado en el procesamiento de imágenes Jaramillo Ortiz, A., Jiménez M., R., & Ramo, O [6] en el año 2014, desarrollado en la Universidad Distrital Francisco José de Caldas en Colombia, plantea como problema que las empresas manufactureras de tuercas y tornillos no tienen un sistema de calidad para dimensiones, por lo cual ellos plantean un sistema de calidad el cual consta de una banda transportadora para la alimentación del sistema; este sistema posee un protocolo de reconocimiento que verifica primero el tipo de pieza, después revisa sus dimensiones, y por ultimo las clasifica en tuercas y tornillos. Este sistema logro automatizar el sistema de calidad con una autonomía de 1400 piezas verificadas y clasificadas por ahora, con una sensibilidad de +/- 0.1 mm.

En Ecuador en el año 2019, Salazar Castillo desarrolló un SISTEMA DE CLASIFICACIÓN Y CONTROL DE CALIDAD EN UN PROCESO DE PRODUCCION INDUSTRIAL USANDO VISION ARTIFICIAL [7]. Este proyecto de doctorado de la Universidad Técnica de Ambato fue desarrollado para una empresa, la cual mostró como sub problemas las tuercas y los tornillos mal terminados con defectos de rebaba y no cumplimiento de las dimensiones, los clientes realizaban gran número de quejas sobre los productos terminados. El problema fue abarcado desarrollando un sistema por medio de una banda transportadora la cual iba conectada a la línea de producción, cuando pasaba por la parte de la cámara, ésta realizaba la medición de las piezas por: dimensiones, color, y problemas en la superficie.

En el proyecto final de doctorado de Mendoza García en 2019, titulado DESGASTE DE PLACAS DE CORTE POR MEDIO DE VISIÓN ARTIFICIAL UTILIZANDO LABVIEW [8] de la Universidad Distrital Francisco José de Caldas, se tiene como problema los sistemas de calidad, por lo tanto, el autor se enfoca en la verificación de las pastillas de corte y su desgaste reflejado en los cortes realizados. El desarrollo de este proyecto es por medio de visión artificial, usando una interfaz gráfica por medio del software LabVIEW. Obtuvo un sistema capaz de medir las pastillas de corte y su vida útil basado en la norma ISO 3685 e ISO 8688 para plaquitas estandarizadas de torneado y fresado respectivamente.

## 4.Marco teórico

### 4.1 Proceso de manufactura:

Los procesos de manufactura es el arte de convertir la materia prima en productos. Los procesos de manufactura están basados en tres partes principales. (1) El diseño del producto, (2) la selección de la materia prima y (3) la secuencia de procesos a través de los cuales será manufacturado el producto [9]. La palabra manufactura viene del latín “manu factus” lo que traduciría hecho a mano, esta involucra la fabricación de productos a través de materias primas usando varios procesos combinando máquinas, operaciones y operadores.

La manufactura es una rama de las ciencias aplicadas bastante compleja, ya que involucra varios recursos y actividades para su buen desarrollo como lo son [9]:

- Diseño del producto.
- Maquinaria y herramienta.
- Planeación de procesos.
- Materiales.
- Compras.
- Control de producción.
- Marketing.
- Embarques.

Entre otras cosas más que requieren los procesos de manufactura. Estos procesos deben ser capaces de responder a varias exigencias y tendencias como lo son [9]:

- Requerimientos del diseño.
- Calidad.
- Flexibles.
- Materiales.
- Métodos de producción e integración por computadora.
- Productividad.

Cuando los procesos de manufactura cumplen con estas exigencias y requerimientos, siempre van a tener un excelente desarrollo y acogida en el mercado.

Los materiales y procesos de manufactura usados en este proyecto serán mencionados a continuación cada uno:

#### 4.1.1 Metales ferrosos:

Los metales ferrosos son aquellos que su principal componente es el hierro [10], este metal cuenta con varias características como lo son la resistencia a la tensión y la dureza. Estos metales pueden ser aleaciones las cuales varían sus componentes de carbono, estaño, plata, platino, magnesio, entre otras. Su composición da propiedades diferentes, para ser usados en diferentes procesos.

#### 4.1.2 Forja:

El forjado es el proceso en el cual el metal es comprimido entre el martillo y un yunque [10], este proceso consiste en la compresión del acero a tal punto que este supere el esfuerzo de fluencia del metal. En su inicio este proceso era artesanal, cuando empezaron las producciones en masa y forjas de piezas más grandes se sustituyó el martillo por un mazo, y el yunque por matrices (o dados) las cuales pueden tener formas variadas dependiendo de las necesidades que se tengan. El mazo puede ser impulsado por medio de potencia mecánica, hidráulica o vapor.

Los procesos de forjado se pueden realizar con las piezas en caliente o en frío [10]. Cuando se forja en caliente las matrices son más productivas ya que el desgaste es menor y se ve reflejado en la cantidad de piezas producidas con mayor calidad. Cuando se forja en frío se tiene más limitaciones, ya que se requiere más potencia en el mazo, a su vez en las matrices aumenta el desgaste y las limitaciones de las piezas a producir.

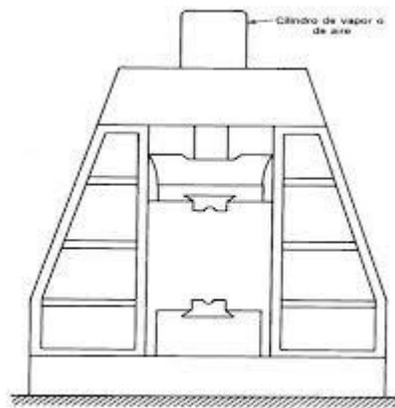
El forjado está clasificado en cuatro:

- **Forjado de herrero:** El forjado de herrero es el arte del herrero por medio de un martillo y un yunque darle forma a una pieza.



*Ilustración 1 Forja de herrero.[10]*

- **Forjado de martinete:** El forjado por martinete es la evolución del forjado de herrero, en donde el herrero empieza con el uso de herramientas como lo es el martillo mecánico, hidráulico o de vapor. Este forjado requiere de matrices para forja libre o forja de estampa.



*Ilustración 2 Forja con martinete.[10]*

- **Forjado en prensa:** Este forjado requiere de una prensa ya sea hidráulica o mecánica, éste también puede ser un forjado libre o estampado.
- **Forjado por recalado:** Este forjado requiere de una recaladora, la cual funciona como una prensa, pero trabaja vertical y horizontalmente a la vez, sujetando el material por un lado y recalándolo por el otro; este forjado está enfocado a la producción de cabezas de tornillos.

#### 4.1.3 Esmerilado:

El esmerilado es un proceso abrasivo [11] en el cual se eliminan o adecuan áreas de materiales por medio de la remoción de material. Este proceso de corte ocurre cuando entra en contacto el material con el frente de la rueda de esmeril. Las ruedas del esmeril son partículas abrasivas de materiales aglutinantes combinadas con diferentes granos para lograr la dureza, resistencia al desgaste, y la tenacidad. Estas dos últimas características son las más importantes cuando se usa como herramienta de corte.



*Ilustración 3 Piedra de esmeril.[11]*

#### 4.1.4 Granallar:

El granallado es un proceso usado para quitar las impurezas en la superficie de los metales, este proceso es usado cuando se va a revestir el metal con anticorrosivos [12], este proceso está dado por el contacto entre dos superficies siendo la primera el acero de la pieza producida o material a trabajar, la segunda es un agente de granallado, el cual puede ser aceites, grasas, sustancias oleosas o similares al aceite. Los agentes buscan transferirse al material granallado dejando la superficie en perfecto estado para otro proceso.



*Ilustración 4 Piezas granalladas. [Autor]*

#### 4.1.5 Perforado:

El perforado es cuando una herramienta de perforación como una broca, gira en torno a su eje longitudinal [13] y se desplaza hacia la pieza que se va a mecanizar, este proceso se puede realizar de diversas formas, la primera es manualmente por medio de un taladro y con fuerza muscular. También se puede realizar por medio de máquina herramienta en la cual la perforación es controlada por medio de lo programado en la máquina. Pero estos procesos funcionan bajo el mismo proceso.

#### 4.1.6 Biselado:

El biselado es un proceso usado para preparar los bordes en las piezas que son móviles en la maquinaria, esta preparación se hace por medio de un corte oblicuo de los bordes [14], este corte se puede hacer por medio de varios procesos como lo son el corte térmico, corte en frío, corte mecánico o corte abrasivo.

#### 4.1.7 Ranurado:

El proceso de ranurado se da cuando se mecaniza una pieza dándole un ancho y una profundidad [15] a lo largo de ésta, existen varios tipos de ranurados y diversas técnicas para hacerlo entre ellas, ranurado múltiple, ranurado y torneado, mecanizado en rampa, ranurado de un solo corte, ángulos achaflanados, torneado con avance axial. para llevar a cabo este proceso se pueden utilizar varias máquinas como lo son seguetas, centros de mecanizados, tornos, entre otros.

#### 4.1.8 Tratamiento térmico:

El tratamiento térmico es el conjunto de procesos los cuales se le aplican a un material con el fin de cambiar sus características iniciales [16]. El tratamiento térmico se puede usar para cambiar las siguientes características:

- Resistencia al desgaste: Capacidad para no erosionar cuando está en contacto con otro material.
- Tenacidad: Es la capacidad que tiene un material de absorber energía sin producir fisuras.
- Maquinabilidad: Capacidad de un material para dejarse mecanizar.
- Dureza: capacidad que tiene un material para dejarse penetrar o rayar.

Los tratamientos térmicos se realizan por medio de dos procesos fundamentales, los cuales son el calentamiento de la pieza y enfriamiento. Cuando se realizan este proceso se realiza con unas pautas, tiempos y temperaturas exactas. Los tratamientos térmicos más usados son:

- Templado: Se usa cuando se le quiere aumentar la dureza a las piezas, este proceso se realiza con una temperatura entre 800°C y 950°C, el enfriamiento debe ser rápido.
- Revenido: Se usa para normalizar los efectos del templado, con esto logran distribuir el templado por toda la pieza. Las temperaturas y el enfriamiento varían dependiendo de cómo se quiera normalizar la pieza.
- Recocido: Se usa cuando se quiere aumentar la elasticidad de la pieza, o su maquinabilidad, se realiza calentando la pieza entre 800°C y 925 °C, después se deja enfriar lentamente.
- Normalizado: Busca dejar el material en su estado normal, sin tensiones internas, como trabajo previo a cualquier otro tratamiento térmico.

#### 4.1.9 Grateado:

El grateado es el proceso en el cual se prepara la superficie [17], se realiza por medio de un cepillo de alambre el cual gira a altas velocidades, la pieza se debe apoyar sobre este para lograr obtener una pieza pulida en la cual se pueden anclar fácilmente diversos recubrimientos.



*Ilustración 5 Cepillo de grateado.[17]*

## 4.2 Sistema de unidades:

Los sistemas de unidades son conjuntos de medidas, los cuales se usan para saber dimensiones como: distancias, pesos, cantidades entre otras. Las medidas se pueden dar en los siguientes sistemas de unidades [18]:

- Sistema cegesimal (sistema científico): Fue establecido por el congreso de electricidad en 1881, se dijo que las magnitudes fundamentales serian: Longitud, masa y tiempo, y estas se debía dar en centímetro, gramo y segundos.
- Sistema Giorgi (sistema práctico): Fue propuesto en 1901 por Giorgi, éste establece las mismas medidas fundamentales que en el cegesimal, pero éstas se debían dar en: metro, kilogramo, y segundo.
- Sistema técnico (terrestre): Este sistema también llamado como el sistema de los ingenieros propone como unidades fundamentales: metro, kilogramo-fuerza, y segundos. Este sistema es muy usado en la ingeniería mecánica.
- Sistema internacional: Fue establecido en 1960 en París, Francia. Este sistema estableció seis medidas o magnitudes fundamentales: longitud, masa, tiempo, intensidad de corriente, temperatura térmica, cantidad de sustancia e intensidad luminosa. Sus unidades son: metro, kilogramo, segundo, amperio, kelvin, mol y candela.

## 4.3 Filtro Gaussiano:

Los filtros Gaussianos son usados para suavizar y eliminar ruidos de la imagen, este filtro se usa para el análisis de imágenes, ya que define mejor los pixeles de la imagen. Este filtro está basado en la función gaussiana de 2 – dimensiones, el resultado de la función se ubica en una matriz [19].

La función gaussiana que rige el filtro es:

$$G_{\sigma}(x, y) = \exp\left(-\frac{x^2+y^2}{2\sigma^2}\right)$$

**Ecuación 1**

Al realizar esta ecuación se debe tener en cuenta que  $\sigma > 0$  . Es un parámetro que denota la desviación estándar de la matriz se debe tener en cuenta que la componente (0,0) es la que tiene mas importancia, los datos que se alejen de esa componente tienden a ser 0.

Al realizar el suavizado de la imagen, se debe aplicar también un normalizado para que se realice en toda la imagen de igual manera, y quede con el mismo rango en todos los pixeles, con el fin de evitar aclarar u oscurecer la imagen. Este normalizado y suavizado se hace por medio de la siguiente ecuación, se debe tener en cuenta el factor que se da.

$$\bar{G}_\sigma(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2+y^2}{2\sigma^2}\right) \quad \text{Ecuación 2}$$

Esta es la expresión que rige el filtro Gauss, la cual debe ser implementa, para llena la matriz asociada H, esta matriz debe cumplir:  $\sum_i \sum_j H(i, j) = 1$  , para así lograr una matriz definida.

#### 4.4 Filtro Canny:

Canny es un filtro usado para la identificación de bordes existentes en una imagen, este se puede programar mediante metodologías diferentes, pero siempre da como resultado que es el mejor filtro para la detección de bordes. Las imágenes son tratadas mediante pixeles y un ambiente discreto, para así poder ver los cambios en las zonas de contorno en la escala de grises [20].

El filtro Canny está basado en tres criterios para la identificación de los bordes:

- Detección: El hecho de evitar la eliminación de bordes importantes y no suministrar falsos bordes.
- Localización: Minimizar la distancia entre la posición real y la localización del borde.
- Respuesta: Integrar las respuestas múltiples, que correspondan a un solo borde.

El algoritmo de Canny está basado en tres pasos:

- Obtención del gradiente:

Cuando se desea obtener el gradiente se debe tener la imagen suavizada y eliminar el ruido mediante un filtro gaussiano, este da como resultado el gradiente de la imagen, se obtiene la magnitud de cada pixel y su módulo de orientación.

- Supresión no máxima al resultado del gradiente:

En el paso anterior se obtuvo una imagen con los bordes definidos pero gruesos, lo que generaría un ruido. En este paso se debe obtener la dirección de cada pixel que más se aproxime a la del gradiente obtenido en el paso anterior. La entrada a este algoritmo es la imagen anterior, con su máscara. Y la salida del algoritmo sería la magnitud  $E_m$  con su orientación del gradiente  $E_o$ .

$$G[f(x, y)] = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial}{\partial x} f(x, y) \\ \frac{\partial}{\partial y} f(x, y) \end{bmatrix} \quad \text{Ecuación 3}$$

$G$  está en dirección de la variación máxima de  $f$  en el punto  $(x, y)$ . Teniendo en cuenta que la magnitud y dirección está dada por:

$$|G| = \sqrt{G_x^2 + G_y^2} = |G_x| + |G_y| \quad \text{Ecuación 4}$$

$$\varphi(x, y) = \tan^{-1} \frac{G_y}{G_x} \quad \text{Ecuación 5}$$

Con estas expresiones calculamos  $E_m$  a partir de la magnitud del gradiente y  $E_o$  por medio de la orientación.

- Histéresis de umbra a la supresión no máxima:

La imagen obtenida en el paso anterior, tiene ruido en la demarcación de los bordes, con este paso se busca eliminar por medio de la histéresis del umbral a la supresión no máxima busca eliminarlos falsos contornos de las imágenes.

- Histéresis de umbral:

La imagen obtenida en el paso anterior, tiene ruido en la demarcación de los bordes, con este paso se busca eliminar por medio de la histéresis del umbral, para así obtener solo los bordes necesarios sin tener en cuenta las uniones de Y, y los segmentos que confluyan en un punto.

#### 4.5 Sistema de visión artificial

Las técnicas de visión artificial requieren tener los siguientes componentes para la realización de sistemas con visión artificial. Estos componentes varían dependiendo de la necesidad que se tenga y el uso del sistema [21].

- Iluminación:

La iluminación en los sistemas de visión artificial se usa para obtener mejores condiciones de la imagen para así mejorar el análisis de ésta, cuando se ubica bien la luz ayuda con la separación de características de la imagen del fondo.

- Óptica:

La óptica es usada como un adicional a la cámara, encargada de captar la luz del exterior y focalizarla en la imagen que será tomada. Esto se ve reflejada en la imagen con baja distorsión, sin errores de perspectiva. Dependiendo de la necesidad que se tenga debe seleccionar el lente a usar.

- Cámaras:

La cámara es usada para capturar imágenes que posteriormente serán usadas como si fuera un sensor, que posteriormente será transmitida a un dispositivo externo. Estas cámaras deben ser seleccionadas dependiendo de la necesidad que se tenga, ya que varía la definición rapidez, que se requiera en el proceso.

- Cables de conexión:

Los cables son una pieza fundamental en los sistemas de visión artificial ya que estos son los encargados de transmitir la imagen tomada por la cámara al procesador. Se debe obtener cables de acuerdo a la necesidad, ya que tener unos cables no apropiados generaría ruido en la imagen lo que dificultaría el tratamiento de ésta.

- Software:

El software es una pieza fundamental en el tratamiento de la imagen ya que es el encargado de extraer y trabajar la información obtenida de la imagen por medio de algoritmos. La gran mayoría de tratamientos usados en los softwares es por medio de librerías de programación.

- Sistemas compactos:

Los sistemas compactos tienen la función autónoma de procesar y tomar decisiones basados en la información entregada por el software. En estos sistemas están conectadas las cámaras o subsistemas del proceso, además pueden ser tarjetas pequeñas o computadores dependiendo de las necesidades que se tengan en el problema.

En la siguiente ilustración se observa los componentes nombrados anteriormente.

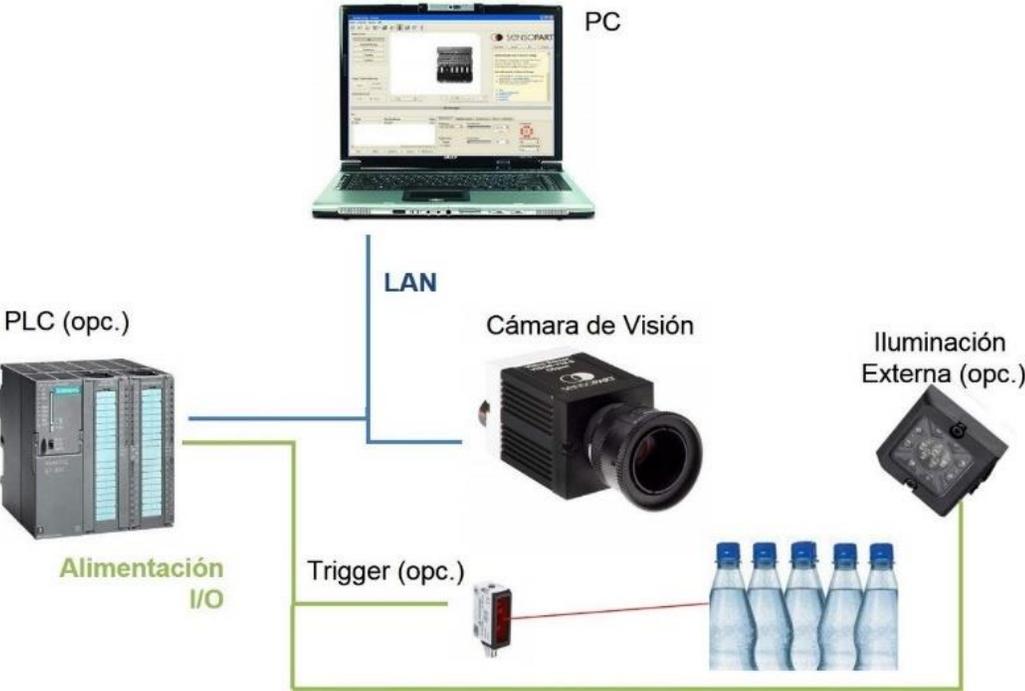


Ilustración 6 Sistema visión artificial.[21]

## 5. Metodología

El proyecto se desarrolló por medio de una metodología en V, en donde la columna descendente está relacionada con las especificaciones y los pasos para el diseño. La parte ascendente está relacionada con la validación de los sistemas, hasta lograr el acoplé funcional de todos los sistemas.

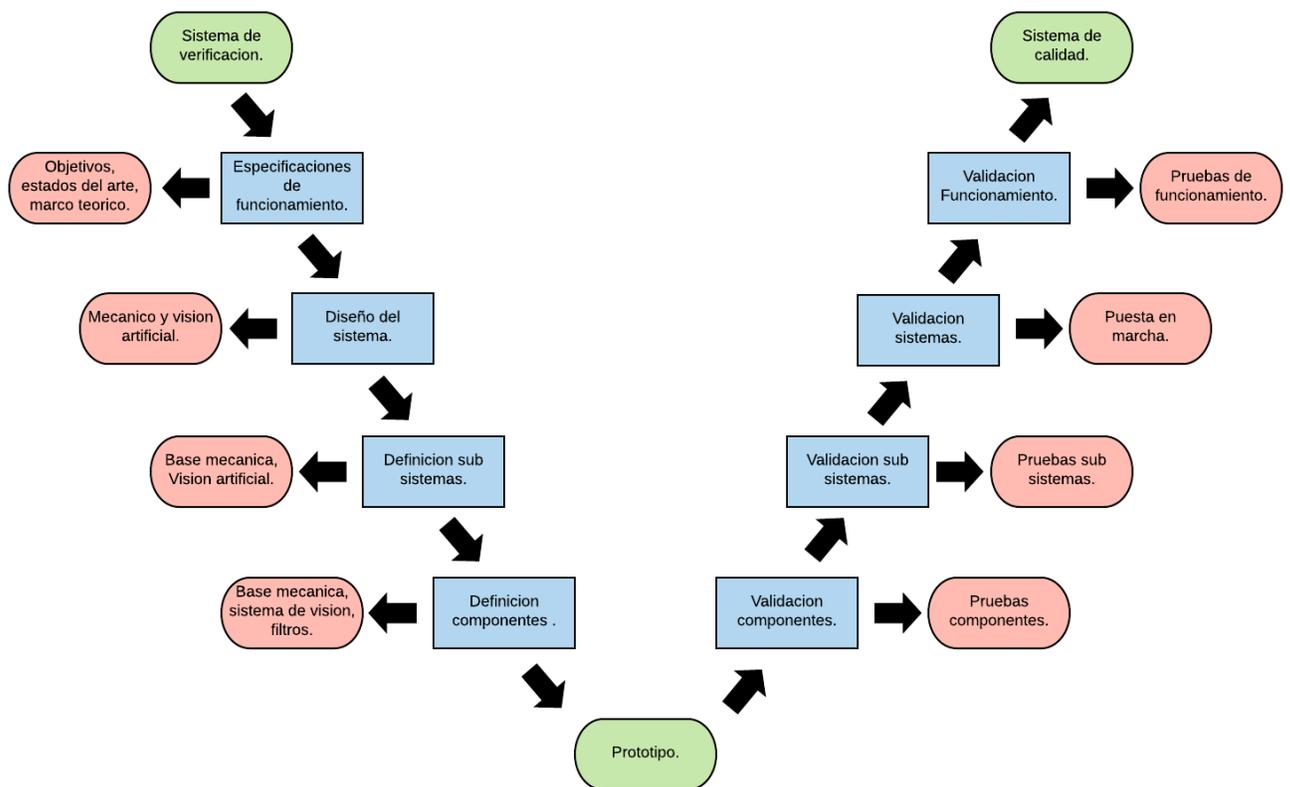


Ilustración 7 Metodología en V.[Autor]

## 6. Desarrollo del proyecto

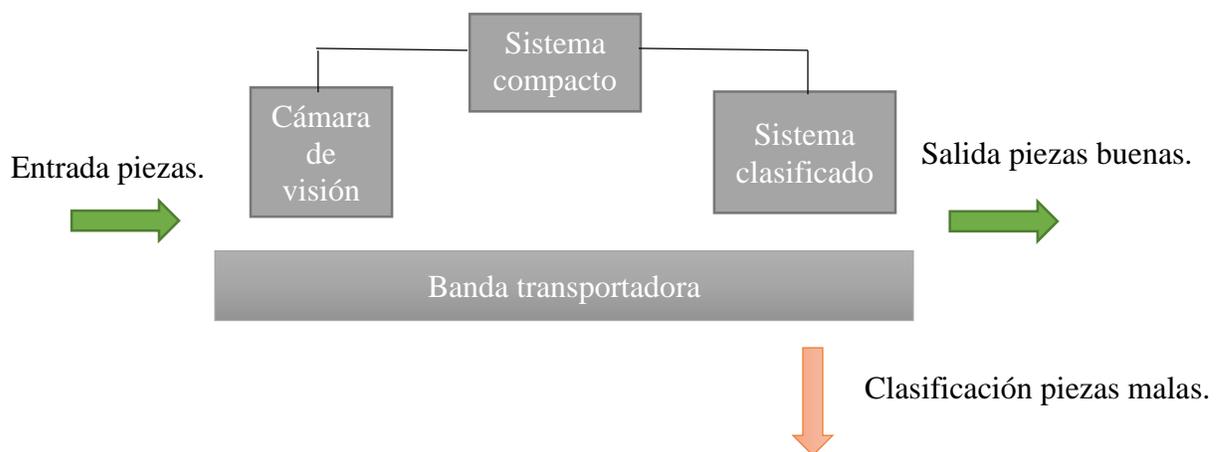
Para el desarrollo del proyecto se realizaron seis etapas:

1. Levantamiento de los requerimientos del equipo.
2. Parametrización de la pieza.
3. Selección.
4. Desarrollo visión artificial.
5. Circuitos eléctricos.
6. Cálculos banda transportadora.

Cumpliendo con las seis etapas de desarrollo se obtuvo un prototipo de un equipo de medición dimensional para piezas del sector agrícola, el cual se desarrolló en conjunto con la empresa Aceros Markert S.A.S.

### 6.1 Levantamiento de los requerimientos del equipo

El levantamiento de requerimientos del equipo se realizó por medio de un bosquejo en el cual se muestran los subsistemas del prototipo.



*Ilustración 8 Bosquejo del sistema.[Autor]*

El equipo requiere tres subsistemas principales, el primero es la banda transportadora encargada de llevar las piezas a los otros subsistemas, el ingreso de las piezas se realiza por el lado izquierdo, avanzado hasta el siguiente subsistema que es la cámara de visión está encargada de obtener los datos de las piezas, y el segundo subsistema es el clasificado de las piezas el cual está encargado de clasificarlas como correctas o incorrectas este está ubicado al final de la banda transportadora. Los subsistemas esta conectados por medio de un sistema compacto que es el encargado de codificar toda la información.

## 6.2 Parametrización

La parametrización de la pieza de la pieza se realizó después de la selección que realizó la empresa Aceros Markert S.A.S; ésta seleccionó un repuesto de maquinaria agrícola el cual es un protector para las cosechadoras John Dear [22].



*Ilustración 9 Guarda. [Autor]*

Para esto la empresa facilitó los planos de la pieza e indicó cuales son las medidas a tener en cuenta para el proceso de clasificado.

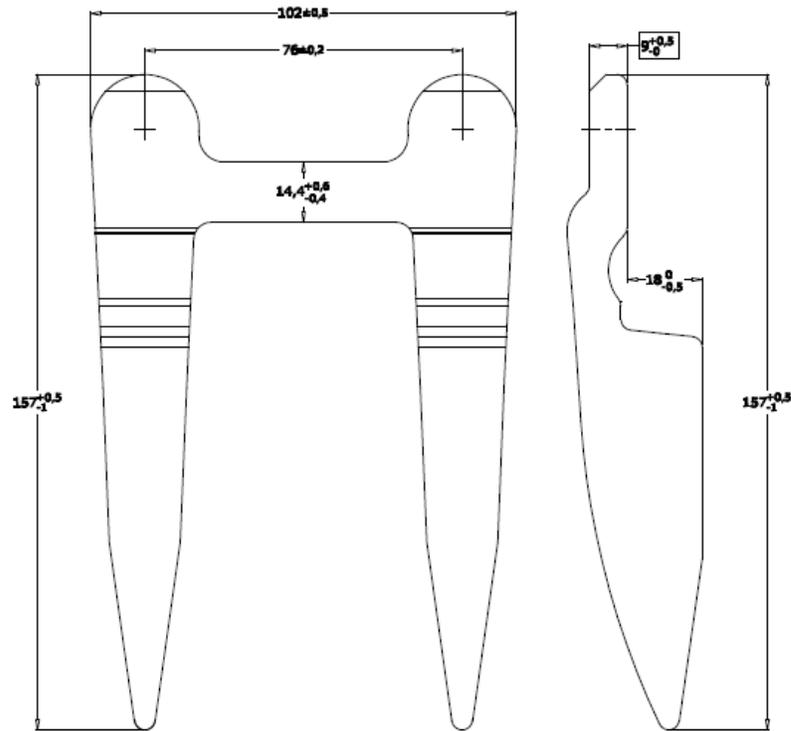


Ilustración 10 Planos guarda.[Autor]

Teniendo el plano de la pieza, se parametrizó un peso promedio del protector para la máquina cosechadora John Dear, el cual tiene un valor de 560 g. Se evaluará el largo y ancho de la pieza, los cuales son de 157 mm y 102 mm, respectivamente.

### 6.3 Selección

La selección de los artículos o elementos requeridos para el desarrollo del proyecto se escogieron mediante una matriz QFD, como se especifica a continuación.

### 6.3.1 Entornos de desarrollo

Después de la revisión del estado del arte y ver que la gran mayoría de los proyectos usaban como entorno de desarrollo Matlab, Labview, y Visual code, se definieron características para compararlos en la matriz QFD.

	Valor Criterio	Matlab	Labview	Visual Code
Portátil	5	1	1	9
Compatibilidad	4	9	3	9
Librerías	4	9	9	9
Simplicidad	3	9	3	3
Total		104	62	126

Tabla 1 Matriz QFD entorno de desarrollo.[Autor]

La matriz QFD arrojó como la mejor opción el **visual code**, por tal razón se desarrolló el código en este entorno de desarrollo.

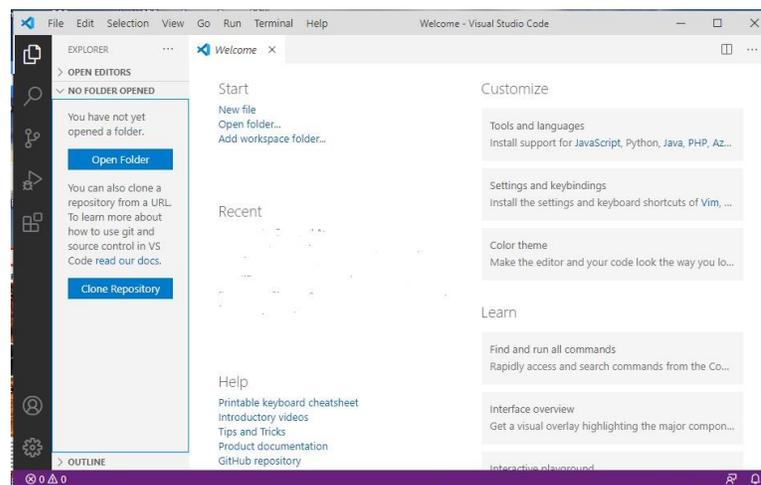


Ilustración 11 Visual code.[Autor]

### 6.3.2 Hardware

Para la selección del hardware se pueden usar tres dispositivos para la realización de estos programas teniendo en cuenta la anterior selección de entorno de desarrollo.

	Valor Criterio	Plc	Arduino	Raspberry
Compatibilidad	5	9	3	9
Capacidad	5	9	3	9
Ejecución	4	9	1	9
Simplicidad	2	3	1	3
Costo	1	1	9	3
Total		133	45	135

Tabla 2 Matriz QFD hardware[Autor].

La tarjeta **Raspberry** fue seleccionada como la mejor opción para usar como hardware del proyecto.



Ilustración 12 Raspberry.[Autor]

### 6.3.3 Cámara de visión

La selección de la cámara se realizó teniendo en cuenta los productos que se encontraban con los proveedores nacionales.

	Valor criterio	C922	C920	C930	C525
Compatibilidad	5	9	9	9	9
Resolución	4	9	9	9	3
Simplicidad	3	3	3	1	9
Stock	2	3	3	3	9
Costo	1	1	3	1	9
Total		97	99	91	111

Tabla 3 Matriz QFD cámara de visión.

La cámara **Logitech C525** fue seleccionada para la toma de datos por visión artificial, con las prestaciones que esta cámara posee.



Ilustración 13 Cámara Logitech C525.[Autor]

### 6.3.4 Arranque del motor

Para la selección del arranque del motor de la banda transportadora se tuvo en cuenta ciertas características de la banda, para el mejor funcionamiento del sistema en conjunto.

	Valor criterio	Variador	Contactores
Control de velocidad	5	9	1
Simplicidad	3	3	9
Costo	1	1	9
Total		55	41

Tabla 4 Matriz QFD arranque motor.

El arranque del motor quedo seleccionado por medio de la matriz QFD, el uso del **variador**.



Ilustración 14 Variador.[Autor]

### 6.3.5 Sistema de clasificación

El sistema de clasificado de las piezas, se realizó la matriz QFD, entre un sistema mecánico, un sistema neumático, y por ultimo un motor lineal.

	Valor criterio	Motor lineal	Sistema mecánico	Neumático
Montaje fácil	4	9	1	3
Compatibilidad	3	9	9	9
Simplicidad	3	9	3	3
Costo	1	1	9	3
Total		91	49	51

*Tabla 5 Matriz QFD sistema de clasificado.*

La matriz QFD, arrojo como mejor resultado el **motor lineal** para la clasificación de las piezas.



*Ilustración 15 Motor lineal.[Autor]*

### 6.4 Banda Transportadora

Para el desarrollo de la banda transportadora se realizaron dos pasos para la elaboración de esta, el primer paso fue los cálculos para elaborar el segundo paso, el cual es el diseño asistido de ésta.

#### 6.4.1 Cálculos.

Para el desarrollo de los cálculos de la banda transportadora se deben definir las dimensiones de la banda: largo de la banda 1.05 [m], ancho de la banda 0.15 [m]. Para continuar con los cálculos se debe tener en cuenta el peso de las piezas y cantidad de piezas que caben en la banda.

$$\text{Peso total} = \text{peso} * \text{cantidad} \quad \text{Ecuación 6}$$

$$\text{Peso total} = 0.56 * 10$$

$$\text{Peso total} = 5.6 \text{ kg}$$

$$\text{Largo} * \text{Ancho} = \text{area banda} \quad \text{Ecuación 7}$$

$$1.05 * 0.15 = 0.1575 \text{ m}^2$$

$$\text{Largo piezas} * \text{ancho} = \text{area piezas en banda} \quad \text{Ecuación 8}$$

$$0.9 * 0.15 = 0.135 \text{ m}^2$$

Se procede a determinar la carga por producto acumulado.

$$M_p = M * F_p * \frac{\text{Porcentaje de area de la banda}}{100} \quad \text{Ecuación 9}$$

$$M_p = \text{Carga por producto acumulado}$$

$$M = \text{Carga del producto}$$

$$F_p = \text{Coeficiente de friccion (Producto banda)}$$

Se procede a calcular la carga del producto.

$$M = \frac{5.6}{0.135} \quad \text{Ecuación 10}$$

$$M = 41.48 \text{ kg/m}^2$$

Se determina el coeficiente de fricción por medio de tablas.

Tabla 3 ( $F_p$ ) COEFICIENTE DE FRICCIÓN EN FUNCIONAMIENTO ENTRE EL ENVASE Y LA BANDA <sup>a</sup>								
MATERIAL DEL ENVASE	MATERIALES ESTÁNDAR <sup>b</sup>							
	POLIPROPILENO		POLIETILENO <sup>c</sup>		ACETAL		ACETAL EC	
	HÚMEDO	SECO	HÚMEDO	SECO	HÚMEDO	SECO	HÚMEDO	SECO
Vidrio	0,18	0,19	0,08	0,09	0,13	0,14	0,13	0,14
Acero	0,26	0,32	0,10	0,13	0,13	0,13	0,19	0,20
Plástico	0,11	0,17	0,08	0,08	0,13	0,16	0,13	0,16
Cartón	—	0,21	—	0,15	—	0,18	—	0,18
Aluminio	0,40	0,40	0,20	0,24	0,33	0,27	0,33	0,27

**Nota:** Las bandas que funcionan en seco en un sistema transportador con acumulación de producto pueden, según la velocidad y el peso, desgastar una superficie dura de la banda, lo que podría incrementar de forma sustancial el coeficiente de fricción.

*Ilustración 16 Coeficientes de fricción. [23]*

Se determinó el coeficiente de fricción como el contacto entre acero, y polipropileno, de un valor de 0.26. Se procede a calcular el porcentaje de banda ocupado.

$$\text{porcentaje banda} = \frac{0,9}{1,05} \quad \text{Ecuación 11}$$

$$\text{porcentaje banda} = 85,7\%$$

Teniendo los datos anteriores se remplazan los valores en la ecuación 9, para calcular por producto acumulado.

$$M_p = 41,48 * 0,26 * 0,857$$

$$M_p = 9,242 \text{ kg/m}_2$$

Se procede a calcular la tracción de la banda.

$$BP = [(M * 2W) * F_w + M_p] * L \quad \text{Ecuación 12}$$

$$w = \text{peso de la banda}$$

$$F_w = \text{Coeficiente de fricción(Guia y banda)}$$

$$L = \text{longitud del transportador}$$

Tabla 2 ( $F_w$ ) COEFICIENTE DE FRICCIÓN DEL ARRANQUE ENTRE LA GUÍA DE DESGASTE Y LA BANDA										
MATERIAL DE LA GUÍA DE DESGASTE	MATERIALES ESTÁNDAR <sup>a</sup>									
	POLIPROPILENO				POLIETILENO		ACETAL		ACETAL EC	
	SUPERFICIE LISA		ABRASIVA <sup>b</sup> LISA		SUPERFICIE LISA		SUPERFICIE LISA		SUPERFICIE LISA	
	HÚMEDO	SECO	HÚMEDO	SECO	HÚMEDO	SECO	HÚMEDO	SECO	HÚMEDO	SECO
U.H.M.W.:	0,11	0,13	NR	NR	0,24	0,32 <sup>c</sup>	0,10	0,10	0,10	0,10
H.D.P.E.	0,09	0,11	NR	NR	NR	NR	0,09	0,08	0,09	0,08
Niñón impregnado con molibdeno o silicona	0,24	0,25	0,29	0,30	0,14	0,13	0,13	0,15	0,13	0,15
Acero inoxidable o al carbono con acabado laminado en frío	0,26	0,26*	0,31	0,31*	0,14	0,15*	0,18	0,19*	0,18	0,19*

Ilustración 17 Coeficiente de fricción [23]

El coeficiente de fricción usado es para U.H.M.W con una superficie húmeda, el cual tiene un valor de 0,11. Reemplazar los valores en la ecuación 12.

$$BP = [(41.48 + 2 * 3.151) * 0,11 + 9,242] * 2.10$$

$$BP = 30.445 \text{ Kg/m}$$

Se procede a calcular la fuerza de tracción ajustada.

$$ABP = BP * SF$$

**Ecuación 13**

$$SF = \text{Factor de servicio}$$

Tabla 6 (SF) FACTOR DE SERVICIO		
Arranques sin carga, con carga aplicada gradualmente.		1,0
Arranques frecuentes bajo carga (más de uno por hora)	AGREGAR 0,2	_____
A velocidades superiores a 100 FPM (pies por minuto) (30 metros/min)	AGREGAR 0,2	_____
Transportadores ascendentes	AGREGAR 0,4	_____
Transportadores por empuje	AGREGAR 0,2	_____
		TOTAL
		_____

**Nota:** Para velocidades superiores a los 50 pies/min (15 m/min) en transportadores que arrancan con acumulación de producto, se recomienda usar motores de arranque suave.

Ilustración 18 Factor de servicio. [23]

El factor de servicio de la banda sería uno, ya que solo tendría un arranque. Reemplazar los valores en la ecuación 13.

$$ABP = 30.445 * 1$$

$$ABP = 30.445 \text{ Kg/m}$$

Se debe calcular la resistencia permitida de por la banda, esto lo realizara por medio de la siguiente ecuación.

$$ABS = BS * T * S \quad \text{Ecuación 14}$$

$BS =$  Resistencia nominal de la banda.

$T =$  Factor de temperatura.

$S =$  Factor de resistencia .

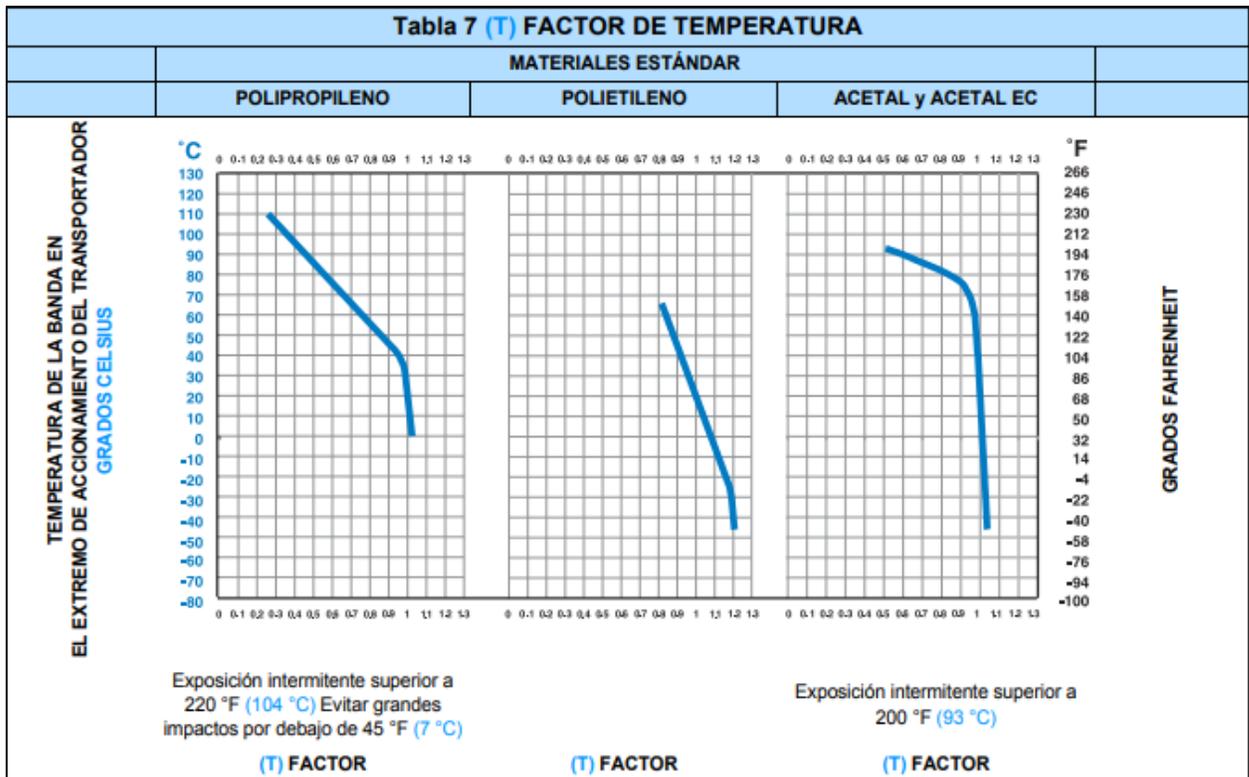


Ilustración 19 Factor de temperatura[23].

$$ABS = 1000 * 0,48 * 0,92$$

$$ABS = 902 \text{ lb/pie}$$

Dado que el ABS es superior al ABP, se puede concluir que las bandas de polipropileno son adecuadas para este trabajo bajo las condiciones de temperatura.

El paso siguiente es el cálculo de la separación máxima entre los engranajes del eje de accionamiento el cual se calcula por medio de la siguiente ecuación.

$$ABSU = (ABP/ABS) * 100\% \quad \text{Ecuación 15}$$

Se procede a calcular la deflexión del eje, por medio de la ecuación 16 en la cual calculamos la carga total del eje y la ecuación 17 la deflexión.

$$w = (ABP + Q) * B \quad \text{Ecuación 16}$$

$$D = \frac{5}{384} * \frac{W * L_S^3}{E * I} \quad \text{Ecuación 17}$$

$E = \text{modulo de elasticidad.}$

$I = \text{momento de inercia.}$

Se procede a calcular el par de torsión máximo, se calcula usando la ecuación 18.

$$T_o = ABP * C * \frac{P.D}{2} \quad \text{Ecuación 18}$$

$P.D = \text{Diametro de paso del engranaje.}$

Por medio de la la ilustración 20, calculamos el valor de A para un eje de acero inoxidable 303 y 304.

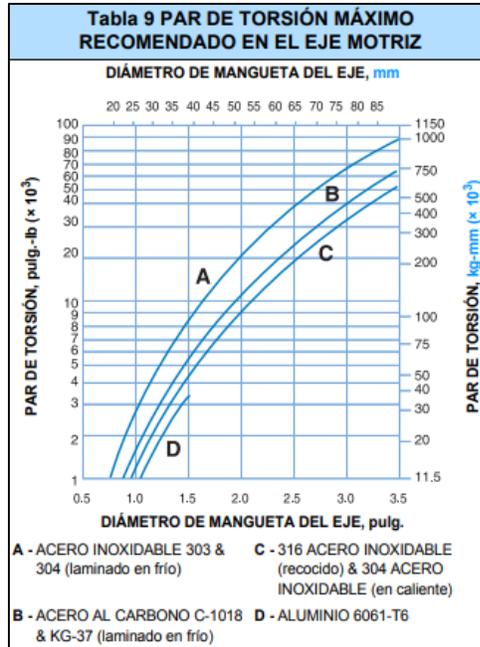


Ilustración 20 Par de torsión.[23].

Se procederá a calcular el valor de  $T_o$  reemplazando los valores en la ecuación 18.

$$T_o = 30.445 * 1.2 * \frac{110}{2}$$

$$T_o = 2009.37 \text{ Kg/mm}$$

Se procede a calcular la potencia de accionamiento de la banda con carga completa por medio de la ecuación 19.

$$\text{Potencia de la banda} = \frac{ABP * C * V}{6,12}$$

**Ecuación 19.**

$$\text{Potencia de la banda} = \frac{30.445 * 1.2 * 6}{6,12}$$

$$\text{Potencia de la banda} = 35.81 \text{ Watts}$$

Teniendo la potencia requerida por la banda se calculará la potencia del motor.

$$Potencia\ del\ motor = \frac{potencia\ banda}{100 - perdidas} * 100 \quad \text{Ecuación 20.}$$

$$Potencia\ del\ motor = \frac{35.81}{100 - 10} * 100$$

$$Potencia\ del\ motor = 39.78\ Watts$$

#### 6.4.2 Diseño Banda

Teniendo en cuenta los cálculos realizados anteriormente se pasó a realizar el diseño de las piezas de la banda teniendo en cuenta las dimensiones de estas, en anexos A se encuentran los planos de la banda, con los cuales se procedió a la construcción del prototipo. Diseño del prototipo quedó como se muestra en la ilustración 21.

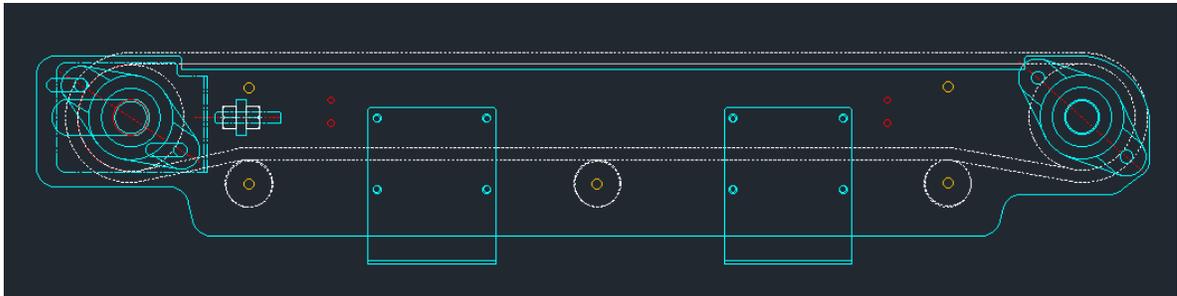
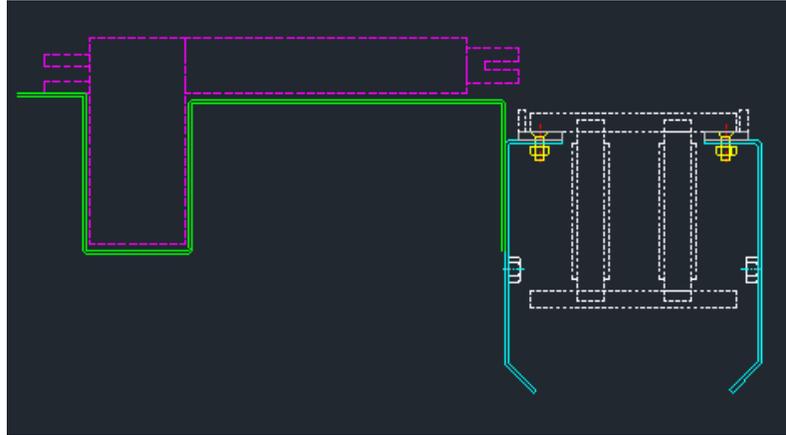


Ilustración 21 Prototipo. [Autor]

El diseño de la banda transportadora para el sistema de medición consta de una banda, en la cual en un extremo tiene un puente para el montaje de la cámara, y en el otro extremo tiene el sistema de clasificación como se muestra en la ilustración 22.



*Ilustración 22 Sistema clasificado.[Autor]*

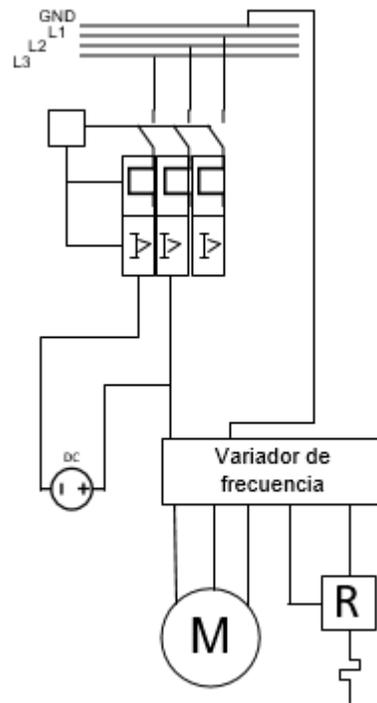
La ilustración 22 es el diseño realizado para el montaje del sistema de clasificación del prototipo del sistema de medición.

## 6.5 Circuitos

El circuito eléctrico del proyecto se dividió en la parte de potencia del proyecto, y la parte de control del proyecto.

### 6.5.1 Circuito potencia.

El circuito de potencia está conectado como se muestra en la ilustración 23, a continuación.

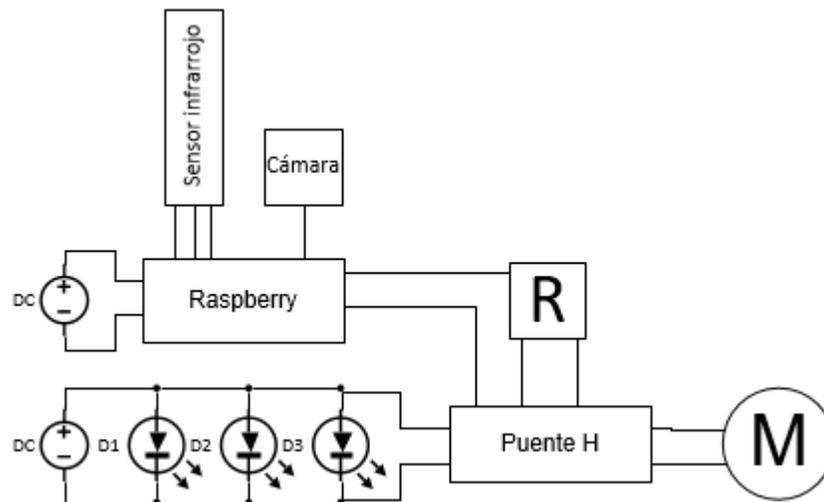


*Ilustración 23 Circuito de potencia.[Autor]*

El circuito de potencia inicia con un guarda motor para el control de las líneas, dos fases se conectan a una fuente de 12 v, otras dos se conectan al variador de frecuencia con la tierra, el variador de frecuencia es controlado por un relé el cual tiene una señal digital de la Raspberry, y el motor va conectado en bajo voltaje y con las tres fases que salen del variador.

### 6.5.2 Circuito de control.

El circuito de control está conectado como se muestra en la ilustración 24, a continuación.



*Ilustración 24 Circuito de control.[Autor]*

El circuito de control está alimentado a 12 V y 5 V, los 12 voltios alimentan la iluminación de la banda, y se conecta el puente H que controla el motor lineal. Los 5 V alimentan la Raspberry, la cual controla el relé de accionamiento del variador de frecuencia y el puente h para controlar el clasificador. El sensor infrarrojo está conectado a la Raspberry para enviar la señal de cuándo debe captar la imagen la cámara.

## 6.6 Visión artificial

### 6.6.1 Conexión Raspberry.

Para realizar la parte de visión se debió empezar con instalarle un sistema operativo Raspberry Pi en la tarjeta SD, para el desarrollo de la herramienta gráfica se debió realizar la configuración de este sistema por medio de los siguientes pasos.

1. Descargar la última versión de Raspberry pi imager.

2. Conectar el lector de tarjeta SD con la Raspberry.
3. Se debe abrir la Raspberry y seleccionar el sistema operativo a trabajar (Descomprimir, Linux).
4. Se transfiere la información a las SD de la Raspberry.

Después de realizar estos pasos se procede a ingresar a la interfaz del sistema operativo para proceder a realizar el código de visión artificial.

### 6.6.2 Código visión.

El código del proyecto funciona bajo el siguiente diagrama de flujo mostrado en la ilustración 25, y el código se encuentra en Anexo B.

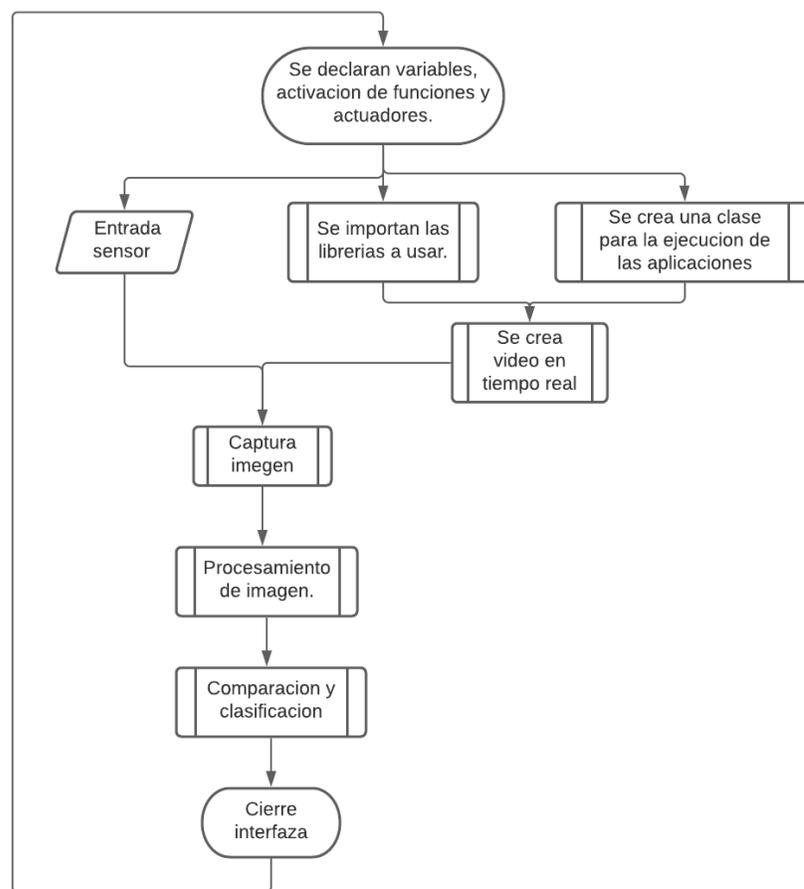


Ilustración 25 Diagrama de flujo. [Autor]

En la ilustración 25 empezar con el código principal el cual es el encargado de llamar las funciones de calcular medidas y activación de los actuadores, posteriormente se procede a realizar la importación de las librerías necesarias para el desarrollo del código. Se procede a definir la clase que ejecutara las funciones de la aplicación.

La función encargada de mostrar el video en tiempo real siempre trabaja de manera paralela con el resto del código, cuando el sensor infrarrojo envía una señal por detección de una pieza ordena tomar una foto y procede a guardarla en la carpeta de trabajo para posteriormente procesarla.

Cuando se procesa la imagen se empieza por cambiar la escala de colores y ponerla en grises para mejorar el procesamiento de esta, posteriormente se procede a aplicar filtros para encontrar los contornos de las piezas, para empezar a evaluar las medidas que se obtiene y clasificarlas para después compara las medidas.

Por último, el programa hace un barrido por los actuadores y sensores para obtener el estado de estos y poder realizar nuevamente un ciclo, cerrando los datos obtenidos anteriormente.

## 7 Equipo construido

La construcción del equipo se realizó en dos pasos, el primero de estos fue la construcción del prototipo todo lo correspondiente a la banda transportadora, el segundo paso fue la conexión entre los subsistemas eléctricos y la parte mecánica.

### 7.1 Construcción parte mecánica.

Se realizó la construcción de la banda, siguiendo los planos dados en el “diseño de la banda”, en este paso se realizó el montaje de la banda con el acople del motor, como se ve en la ilustración 25.

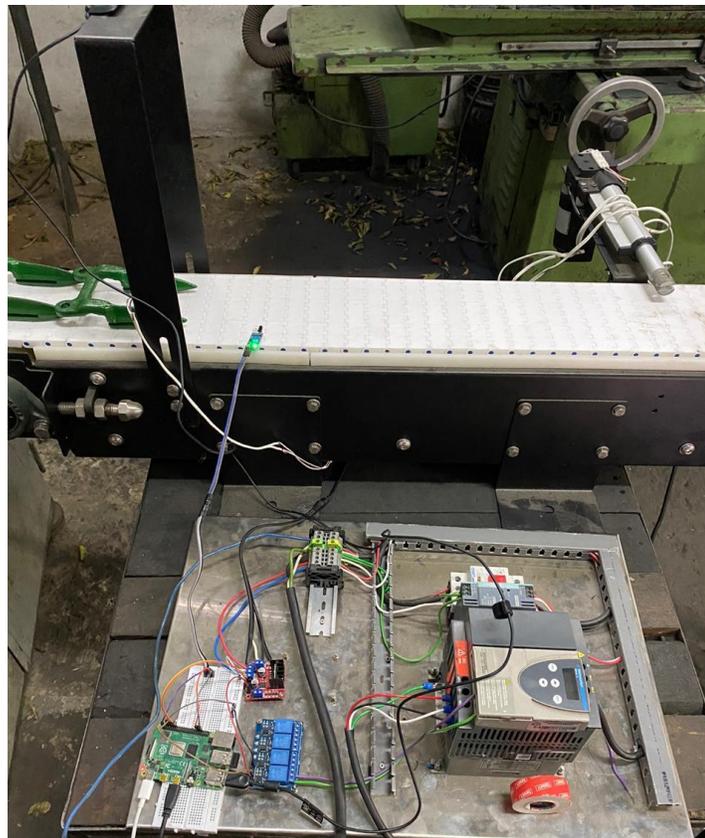


*Ilustración 26 Construcción parte mecánica.[Autor]*

En la ilustración 25 se puede observar la parte mecánica del proyecto la cual cuenta con la banda transportadora acoplada al motor, un puente para montar la cámara, y la iluminación del sistema. Al finalizar la banda se tiene un motor lineal acoplado a ésta para realizar la clasificación del sistema.

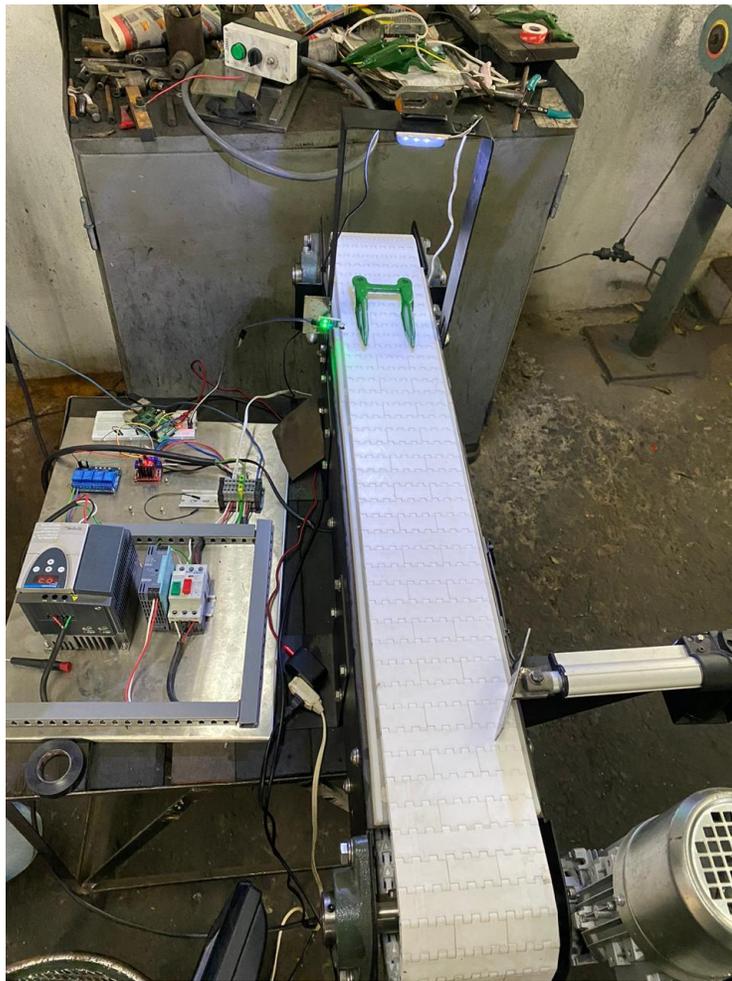
## 7.2 Conexión subsistemas.

Las conexiones de los subsistemas se realizaron mediante los circuitos mostrados en el apartado de circuitos, se realizó la conexión de la parte de potencia, y la parte de control que gobierna la parte mecánica del sistema, en la ilustración 26 se ve la conexión de los subsistemas.



*Ilustración 27 Conexión subsistemas.[Autor]*

Las conexiones de los subsistemas se hicieron siguiendo los planos eléctricos, se realizó la configuración del variador con los parámetros del motor, y se configuro a una frecuencia de 15 Hz. Se realizó la unión de tierras entre la Raspberry, y la fuente de 12v, se configuro lo Raspberry para controlar el relé que controla el variador, y el puente H que controla el motor lineal encargado de clasificar las piezas.



*Ilustración 28 Prototipo.[Autor]*

## 8 Pruebas y Resultados

De un lote de 1100 piezas se seleccionaron 100 piezas para la realización de las pruebas, para la validación del sistema de visión artificial, el cual es encargado de la toma de datos y comparación de estos con las medias estandarizadas de la pieza (ver plano), para calcular el porcentaje de error y clasificarlos. Los datos de las pruebas están en el Anexo C.



*Ilustración 29 Pruebas prototipo.[Autor]*

Las pruebas se realizaron en lotes de 10 piezas, las cuales se pasaron por el sistema y posteriormente se verificaban manualmente por medio de un calibrador. El sistema tarda 3 minutos y 34 segundos en validar 10 piezas, cada pieza en promedio se demora 20 segundos en cumplir el ciclo de evaluación.

### 8.1 Largo de pieza:

Se realizaron 100 muestras del largo de pieza, en los cuales se tomaron los datos prácticos arrojados por el sistema, y los teóricos medidos con el calibrador, a continuación, en la tabla 6 se muestran los resultados promedio.

Largo práctico	Largo teórico	Error	Desviación
156,99	157,00	2,62%	0,034

*Tabla 6 Resultados largo de pieza.[Autor]*

En las pruebas del largo de pieza se obtuvo un porcentaje de error del 2.65% y una desviación estándar de 0.034, en 100 pruebas realizadas.

### 8.2 Ancho de pieza:

Se realizaron 100 muestras del largo de pieza, en los cuales se tomaron los datos prácticos arrojados por el sistema, y los teóricos medidos con el calibrador, a continuación, en la tabla 6 se muestran los resultados promedio.

Ancho practico	Ancho teórico	Error	Desviación
102,00	101,98	4,13%	0,035

*Tabla 7Resultados ancho de pieza.[Autor]*

En las pruebas del ancho de pieza se obtuvo un porcentaje de error del 4.13% y una desviación estándar de 0.035, en 100 pruebas realizadas.

### 8.3 Precisión:

La precisión de un sistema se obtiene evaluando la cantidad de veces que el sistema es capaz de obtener el mismo resultado a través de la elaboración de las pruebas. Esta se calcula por medio de la ecuación 21. Se realizaron 10 pruebas para una misma pieza, buscando la precisión que tiene el sistema en el largo y ancho de la pieza.

$$Precision = \frac{Aciertos}{Aciertos+fallos} \quad \text{Ecuación 21}$$

#### 8.3.1 Aciertos largo de pieza:

Se realizaron 10 pruebas para calcular la precisión del sistema en el ancho de la pieza.

Aciertos largo de pieza	Fallos largo de pieza
7	3

Tabla 8 Precisión largo.[Autor]

$$Precision = \frac{7}{10}$$
$$Precision = 0.7$$

Se obtuvo una precisión de 0.7 para el largo de pieza.

#### 8.3.2 Aciertos ancho de pieza:

Se realizaron 10 pruebas para calcular la precisión del sistema en el ancho de la pieza.

Aciertos largo de pieza	Fallos largo de pieza
9	1

Tabla 9 Precisión ancho de pieza.[Autor]

$$Precision = \frac{9}{10}$$
$$Precision = 0.9$$

Se obtuvo una precisión de 0.9 para el ancho de pieza.

## 9 Conclusiones

- Se caracterizó y parametrizó la pieza guarda (referencia) de la cosechadora Jhon Dear de maquinaria agrícola a desarrollar en la empresa Aceros Markert S.A.S.
- Se diseñó un sistema eléctrico de control por medio de la plataforma Visual Code y un sistema de visión artificial por medio del filtro gaussiano y Canny para el control de las medidas del equipo de medición dimensional para piezas del sector agrícola.
- Se construyó el prototipo del equipo dimensional para piezas del sector agrícola articulando el sistema de control y el sistema de visión artificial con la parte mecánica del prototipo, la cual consta de una banda transportadora y un actuador de clasificación. El prototipo de la máquina construida tiene una precisión de 0.7 para el largo de pieza y 0.9 para el ancho de la pieza, estando esta precisión en un rango de 0 a 1.
- Se desarrolló un sistema de medición dimensional para la empresa Aceros Markert S.A.S el cual permitió evaluar 10 piezas en 3 minutos y 34 segundos, y se lograría evaluar el lote de 1100 piezas en aproximadamente 5 horas y 50 minutos. Actualmente esta evaluación se hace de forma manual por un operario y toma alrededor de 11 horas laborales.
- Se obtuvo un prototipo funcional del equipo de medición dimensional para piezas del sector agrícola por medio de visión artificial, el cual tuvo un porcentaje de error del 2.65% con una desviación estándar de 0.034 para la medida del largo. La medida del ancho tuvo un porcentaje de error del 4.13% y una desviación estándar de 0.035; lo cual es aceptable según los requerimientos de la empresa Aceros Markert S.A.S.

## 10 Bibliografía

- [1] Faostat. (10 de diciembre de 2012). obtenido de <http://www.fao.org/faostat/es/#data/qc>
- [2] Martel, p., & paris, d. (2007). Estados unidos patente nº us 2007/0111648 a1 .
- [3] Profesional, e. e. (2012). Aplicación práctica de la visión artificial. el fse.
- [4] Porras , j., & de la cruz, m. (2014). Clasification system based on computer vision. cercado de lima.
- [5] Ostroukh, A., Morozova, T., Nikonov, V., Ivanova, I., Sumkin, K., & Akimov, D. (2014).Development of contactless integrated interface of complex production lines. Moscow.
- [6]Jaramillo Ortiz, A., Jiménez M., R., & Ramo, O. I. (2014). Inspección de calidad para un sistema de producción industrial basado en el procesamiento de imágenes. Bogota, Colombia
- [7] Salazar Castillo, J. e. (2019). Sistema de clasificación y control de calidad en un proceso de produccion industrial usando vision artificial. Ambato ,Ecuador.
- [8] Mendoza García, J. s. (2019). Desarrollo de software para la medición de desgaste de placas de corte por medio de visión artificial utilizando labview . Bogota.
- [9] Sanchez Garcia, G. (2002). Manufactura, ingenieria y tecnologia. Mexico : Prentice hall.
- [10] Eraso Guerrero, O. (2008). Procesos de manufactura.
- [11] Villasenor, E., & Barreras Espinoza, j. (29 de enero de 2013). Esmerilado y procesos abrasivos.
- [12] Pantatec. (18 de diciembre de 2019). obtenido de <https://www.pantatec.de/es/tecnologia/proceso-de-granallado/>
- [13] Emag. (18 de diciembre de 2019). obtenido de <https://www.emag.com/es/tecnologias/perforado.html>
- [14]Mipsa.(19 de diciembre de 2019).obtenido de mipsa:<https://www.mipsa.com.mx/dotnetnuke/procesos/biselado-mecanico>
- [15] Jorge, D. (27 de abril de 2015). obtenido de [https://prezi.com/yqfma\\_igvcuy/el-ranurado/](https://prezi.com/yqfma_igvcuy/el-ranurado/)
- [16] Pazos, N. P. (2006). Tecnologia de los metales y procesos de manufactura . Caracas.
- [17] Vargas Jimenez , W. A. (2017). Propuesta para el mejoramiento del proceso de limpieza y aplicación de recubrimientos para la estructura metalica producida por la empresa jarco s.a. Sogamoso.
- [18] Galan Garcia , J. L. (1987). Sistema de unidades fisicas. Barcelona: reverté.

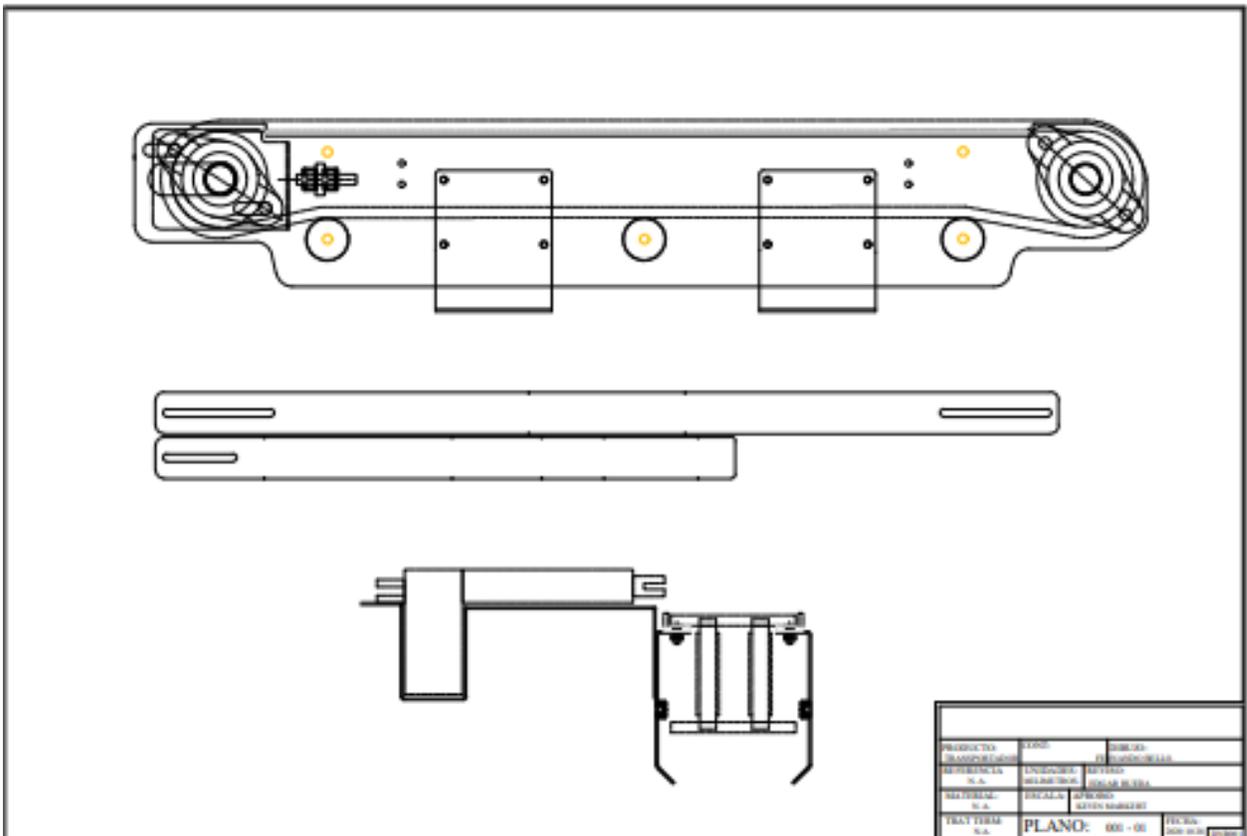
- [19] Diaz De Greñu de Pedro , j. (2014). Analisis matematico de rutinas de procesamiento de imagenes digitales en fiji/imagenj. La Rioja.
- [20] Valverde Rebaza, j. (2007). Detección de bordes mediante el algoritmo de canny. Trujillo .
- [21] Infaimon. (31 de Enero de 2020). Infaimon. Obtenido de <https://www.infaimon.com/>
- [22]Deere. (04 de Agosto de 2020). John Dear. Obtenido <https://www.deere.es/es/cosechadoras/>
- [23] Intralox. (2018). Manual de instalacion, mantenimiento y solucion de problemas de las bandas transportadoras . Seccion 3. pag. 403 - 444.

## 11. Anexos

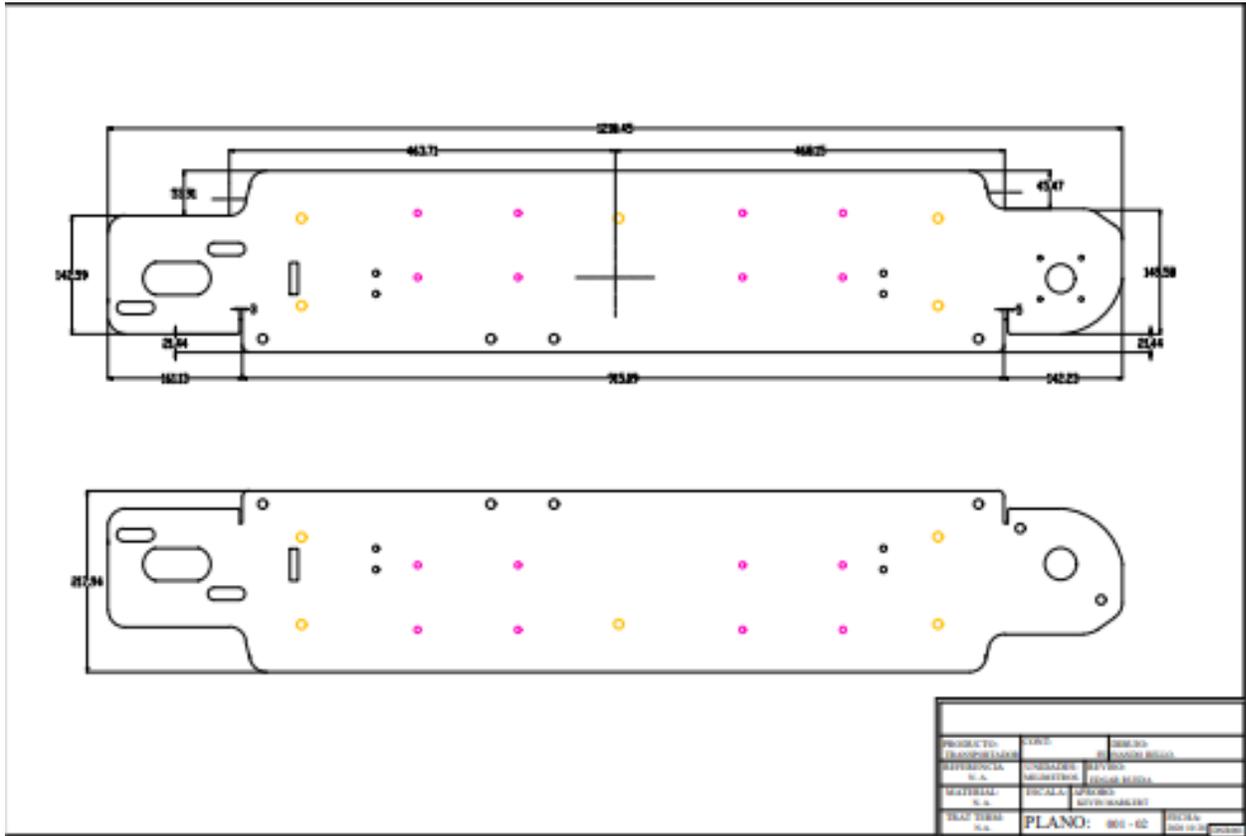
En los anexos encontraron los planos de la banda, y una parte del código usado en el proyecto.

### 11.1 Anexo A

Planos mecánicos para el desarrollo de la banda transportadora:







## 11.2 Anexo B.

El código a usar para el desarrollo del proyecto:

```

from __future__ import print_function
from pyimagesearch.photoboothapp import PhotoBoothApp
from imutils.video import VideoStream
import argparse
import time

# construct the argument parse and parse the arguments
# initialize the video stream and allow the camera sensor to warmup
print("[INFO] warming up camera...")
vs = VideoStream(usePiCamera= 0).start()
time.sleep(2.0)

```

```
# start the app
pba = PhotoBoothApp(vs,r"/home/pi/Documents/Proyec/output")
pba.root.mainloop()
```

#### FUNCIONES:

```
# import the necessary packages
from __future__ import print_function
from PIL import Image
from PIL import ImageTk
import tkinter as tk
import threading
import datetime
import os
from scipy.spatial import distance as dist
from imutils import perspective
from imutils import contours
import numpy as np
import imutils
import cv2
from cv2 import cv2
import RPi.GPIO as GPIO
import time
GPIO.setwarnings(False)
GPIO.setmode(GPIO.BCM)
GPIO.setup(2,GPIO.IN)
GPIO.setup(14,GPIO.OUT)
GPIO.setup(15,GPIO.OUT)
GPIO.setup(18,GPIO.OUT)
```

```
class PhotoBoothApp:
```

```
    def __init__(self, vs, outputPath):
        # store the video stream object and output path, then initialize
        # the most recently read frame, thread for reading frames, and
        # the thread stop event

        self.vs = vs
        self.outputPath = outputPath
        self.frame = ('900x600')
        self.thread = None
        self.stopEvent = None
        # initialize the root window and image panel
        self.root = tk.Tk()
        self.width = self.root.winfo_screenwidth()
```

```

self.height = self.root.winfo_screenheight()
self.root.geometry(str(self.width)+'x'+str(self.height))
self.panel = None
self.activo=True
self.framerresult = tk.Frame(self.root,width=self.width*0.5,height=self.height*0.5)
self.framerresult.pack(side="right")
#self.framerresult.geometry('350x200')

lbl = tk.Label(self.framerresult, text="Hello")
lbl.grid(column=0, row=0)
# create a button, that when pressed, will take the current
# frame and save it to file
"""btn = tk.Button(self.root, text="Capturar la imagen",
    command=self.takeSnapshot,bd=3, relief="groove",font=("Segoe UI",12))
btn.pack(side="bottom", expand="no", padx=1,
    pady=5)

btn1 = tk.Button(self.root, text="Analizar la imagen",
    command=self.detec,bd=3, relief="groove",font=("Segoe UI",12))
btn1.pack(side="bottom", expand="no", padx=1,
    pady=1)"""
# start a thread that constantly pools the video sensor for
# the most recently read frame

self.stopEvent = threading.Event()
self.thread = threading.Thread(target=self.videoLoop, args=())

self.thread.start()

# set a callback to handle when the window is closed
self.root.wm_title("ACEROS MARKERT")
#self.root.config(bg='beige',relief='sunken')
self.root.wm_protocol("WM_DELETE_WINDOW", self.onClose)

self.threada = threading.Thread(target=self.active, args=())
self.threada.start()

def videoLoop(self):
    # DISCLAIMER:
    # I'm not a GUI developer, nor do I even pretend to be. This
    # try/except statement is a pretty ugly hack to get around
    # a RunTime error that Tkinter throws due to threading

```

```

try:
    # keep looping over frames until we are instructed to stop
    while not self.stopEvent.is_set():
        # grab the frame from the video stream and resize it to
        # have a maximum width of 300 pixels
        self.frame = self.vs.read()
        self.frame=self.frame[0:, 180:560]
        self.frame = imutils.resize(self.frame)
        # OpenCV represents images in BGR order; however PIL
        # represents images in RGB order, so we need to swap
        # the channels, then convert to PIL and ImageTk format
        image = cv2.cvtColor(self.frame, cv2.COLOR_BGR2RGB)
        image = Image.fromarray(image)
        image = ImageTk.PhotoImage(image)

        # if the panel is not None, we need to initialize it
        if self.panel is None:
            self.panel = tk.Label(image=image)
            self.panel.image = image
            self.panel.pack(side="left", padx=10, pady=10)

        # otherwise, simply update the panel
        else:
            self.panel.configure(image=image)
            self.panel.image = image

except RuntimeError as e:
    print("[INFO] caught a RuntimeError")

def takeSnapshot(self):
    # grab the current timestamp and use it to construct the
    # output path
    ts = datetime.datetime.now()
    filename = "Analizar.jpg"
    p = os.path.sep.join((self.outputPath, filename))

    # save the file
    self.frame1 = self.frame[self.Xi:self.Xo, self.Yi:self.Yo]
    cv2.imwrite(p, self.frame1.copy())
    print("[INFO] saved {}".format(filename))

def active(self):

```

```

while self.activo == True:

    GPIO.output(14,0)
    GPIO.output(15,0)
    GPIO.output(18,0)
    input_value = GPIO.input(2)
    #print(input_value)
    fFrame = [[0,900,0,380,0],[0,200,0,380,1]]
    medidasFijas = [[167.5,101],[11,11]]
    medidasFoto = []
    if input_value == 0:
        print("12=0")
        GPIO.output(14,1)
        #GPIO.output(24,0)
        for m in fFrame:
            self.Xi = m[0]
            self.Xo = m[1]
            self.Yi = m[2]
            self.Yo = m[3]
            self.fr = m[4]
            self.takeSnapshot()
            time.sleep(1)
            self.detec()
            try:
                ress = [self.dimA,self.dimB]
            except:
                ress = [0,0]
            medidasFoto.append(ress)
        print(medidasFoto)
        print(medidasFijas)
        self.Dlargo = abs(medidasFoto[0][0]-medidasFijas[0][0])
        self.Dancho = abs(medidasFoto[0][1]-medidasFijas[0][1])
        self.Ddiametro = abs(medidasFoto[1][0]-medidasFijas[1][0])
        print(self.Dlargo,self.Dancho,self.Ddiametro)
    if self.Dlargo<=4000:
        GPIO.output(15,0)
        GPIO.output(18,1)
        time.sleep(3)
    if self.Dancho<=40000:
        GPIO.output(15,0)
        GPIO.output(18,1)
        time.sleep(3)
    if self.Ddiametro<=10000:

```

```

        print("correcto")
        GPIO.output(15,0)
        GPIO.output(18,1)
        time.sleep(3)
    else:
        print("diametro incorrecto")
        GPIO.output(15,0)
        GPIO.output(18,1)
    else:
        print("ancho incorrecto")
        GPIO.output(15,0)
        GPIO.output(18,1)
    else:
        print("largo incorrecto")
        GPIO.output(15,0)
        GPIO.output(18,1)

    GPIO.output(14,0)
    time.sleep(3)

```

```

def onClose(self):
    # set the stop event, cleanup the camera, and allow the rest of
    # the quit process to continue
    GPIO.output(14,0)
    GPIO.output(15,0)
    GPIO.output(18,0)
    print("[INFO] closing...")
    #self.thread.set()
    #self.stopEvent.set()
    self.vs.stop()
    self.root.destroy()

```

```

def detec (self):
    self.ddB=[]
    self.ddA=[]
    self.dDB=[]
    self.dDA=[]
    def midpoint(ptA, ptB):
        return ((ptA[0] + ptB[0]) * 0.5, (ptA[1] + ptB[1]) * 0.5)
    try:
        # construct the argument parse and parse the arguments

```

```

width = 103.8
# load the image, convert it to grayscale, and blur it slightly
image = cv2.imread(r"/home/pi/Documents/Proyec/output/Analizar.jpg")
#image = imutils.resize(image, width=600)

gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
gray = cv2.GaussianBlur(gray, (7, 7), 0 )
#gray = cv2.bilateralFilter(gray, 9, 75, 75)
#gray = cv2.medianBlur(gray, 5)

# perform edge detection, then perform a dilation + erosion to
# close gaps in between object edges
edged = cv2.Canny(gray, 50, 100)
#cv2.imshow("GRAY1",edged)
edged = cv2.dilate(edged, None, iterations=1)
#cv2.imshow("GRAY2",edged)
edged = cv2.erode(edged, None, iterations=1)
#cv2.imshow("GRAY3",edged)
# find contours in the edge map
cnts = cv2.findContours(edged.copy(), cv2.RETR_EXTERNAL,
                        cv2.CHAIN_APPROX_SIMPLE)
cnts = imutils.grab_contours(cnts)

# sort the contours from left-to-right and initialize the
# 'pixels per metric' calibration variable
(cnts, _) = contours.sort_contours(cnts)
pixelsPerMetric = None

# loop over the contours individually
print(len(cnts))
for c in cnts:
    # if the contour is not sufficiently large, ignore it
    if cv2.contourArea(c) < 100:
        continue

    # compute the rotated bounding box of the contour
    orig = image.copy()
    box = cv2.minAreaRect(c)
    box = cv2.cv.BoxPoints(box) if imutils.is_cv2() else cv2.boxPoints(box)
    box = np.array(box, dtype="int")

    # order the points in the contour such that they appear

```

```

# in top-left, top-right, bottom-right, and bottom-left
# order, then draw the outline of the rotated bounding
# box
box = perspective.order_points(box)
cv2.drawContours(orig, [box.astype("int")], -1, (0, 255, 0), 2)

# loop over the original points and draw them
for (x, y) in box:
    cv2.circle(orig, (int(x), int(y)), 5, (0, 0, 255), -1)

# unpack the ordered bounding box, then compute the midpoint
# between the top-left and top-right coordinates, followed by
# the midpoint between bottom-left and bottom-right coordinates
(tl, tr, br, bl) = box
(tltrX, tltrY) = midpoint(tl, tr)
(blbrX, blbrY) = midpoint(bl, br)

# compute the midpoint between the top-left and top-right points,
# followed by the midpoint between the top-right and bottom-right
(tlbrX, tlbrY) = midpoint(tl, br)
(trbrX, trbrY) = midpoint(tr, br)

# draw the midpoints on the image
cv2.circle(orig, (int(tltrX), int(tltrY)), 5, (255, 0, 0), -1)
cv2.circle(orig, (int(blbrX), int(blbrY)), 5, (255, 0, 0), -1)
cv2.circle(orig, (int(tlbrX), int(tlbrY)), 5, (255, 0, 0), -1)
cv2.circle(orig, (int(trbrX), int(trbrY)), 5, (255, 0, 0), -1)

# draw lines between the midpoints
cv2.line(orig, (int(tltrX), int(tltrY)), (int(blbrX), int(blbrY)),
         (255, 0, 255), 2)
cv2.line(orig, (int(tlbrX), int(tlbrY)), (int(trbrX), int(trbrY)),
         (255, 0, 255), 2)

# compute the Euclidean distance between the midpoints
dA = dist.euclidean((tltrX, tltrY), (blbrX, blbrY))
dB = dist.euclidean((tlbrX, tlbrY), (trbrX, trbrY))

# if the pixels per metric has not been initialized, then
# compute it as the ratio of pixels to supplied metric
# (in this case, inches)
if pixelsPerMetric is None:
    pixelsPerMetric = dB / width

```

```

# compute the size of the object
self.dimA = dA / pixelsPerMetric
self.dimB = dB / pixelsPerMetric
if self.fr ==0:
    self.dimB = self.dimB-2
if self.fr ==1:
    self.dimA = self.dimA/10
    self.dimB = self.dimB/10

self.ddA.append(self.dimA)
self.ddB.append(self.dimB)
# draw the object sizes on the image
cv2.putText(orig, "{:.1f} [mm]".format(self.dimA),
            (int(tltrX - 15), int(tltrY - 10)), cv2.FONT_HERSHEY_SIMPLEX,
            0.65, (255, 255, 255), 2)
cv2.putText(orig, "{:.1f} [mm]".format(self.dimB),
            (int(trbrX + 10), int(trbrY)), cv2.FONT_HERSHEY_SIMPLEX,
            0.65, (255, 255, 255), 2)

print(len(self.dDA))
self.dimA = round(max(self.ddA), 2)
self.dimB = round(max(self.ddB),2)
# show the output image
#cv2.namedWindow("Image", cv2.WINDOW_KEEPRATIO)
#label = tk.Label(self.framerresult, image=orig, bg="green")
#self.framerresult=cv2.imshow("Image", orig)
#""
cv2.imshow("Image", orig)
wait_time = 1
while cv2.getWindowProperty('Image', cv2.WND_PROP_VISIBLE) >= 1:
    keyCode = cv2.waitKey(wait_time)
    if (keyCode & 0xFF) == ord("q"):
        cv2.destroyAllWindows()
        break
#""

```

```

except:
    self.dimA = 0
    self.dimB = 0

```

### 11.3 Anexo C.

Los resultados obtenidos en las 100 pruebas realizados se encuentran en la siguiente tabla.

Prueba	Largo practico (mm)	Largo teórico (mm)	Error	Ancho practico (mm)	Ancho teórico (mm)	Error
1	156,95	157	3,18%	102,03	101,99	3,92%
2	156,92	157,01	5,73%	102,06	101,99	6,86%
3	156,95	156,99	2,55%	101,94	101,97	2,94%
4	156,95	157,01	3,82%	102,04	101,99	4,90%
5	156,95	157	3,18%	102,06	102,01	4,90%
6	156,94	156,99	3,18%	101,95	101,99	3,92%
7	156,92	157	5,10%	102,07	101,99	7,84%
8	157,03	156,98	3,19%	101,99	102,01	1,96%
9	157,06	157,01	3,18%	102,00	101,96	3,92%
10	156,95	156,99	2,55%	101,93	102,02	8,82%
11	157,06	157	3,82%	102,00	101,98	1,96%
12	156,92	157,02	6,37%	101,94	102,01	6,86%
13	156,94	156,99	3,18%	101,98	102,00	1,96%
14	157,05	157,01	2,55%	102,07	102,00	6,86%
15	156,93	157,01	5,10%	102,02	101,97	4,90%
16	156,99	157,01	1,27%	102,00	101,95	4,90%
17	157,05	156,98	4,46%	102,01	102,01	0,00%
18	156,96	157,01	3,18%	102,05	101,95	9,81%
19	156,98	157	1,27%	102,06	101,98	7,84%
20	156,96	157	2,55%	101,94	101,97	2,94%
21	157,04	157,01	1,91%	102,05	102,00	4,90%
22	157,02	156,99	1,91%	101,96	102,02	5,88%
23	157,02	156,98	2,55%	101,93	102,00	6,86%
24	157,00	157,01	0,64%	101,98	101,97	0,98%
25	156,96	157	2,55%	102,00	102,02	1,96%
26	157,03	157,02	0,64%	102,02	102,00	1,96%
27	156,99	156,99	0,00%	101,95	101,95	0,00%
28	156,94	156,98	2,55%	102,02	102,02	0,00%
29	157,06	157	3,82%	101,96	101,98	1,96%
30	157,01	157,02	0,64%	102,07	101,99	7,84%

31	156,96	157,01	3,18%	101,95	101,98	2,94%
32	156,95	157,01	3,82%	102,05	101,99	5,88%
33	157,05	157,01	2,55%	102,06	101,95	10,79%
34	157,04	156,99	3,18%	102,07	102,02	4,90%
35	157,03	157,01	1,27%	101,94	101,96	1,96%
36	156,94	156,98	2,55%	101,94	101,95	0,98%
37	156,97	157	1,91%	102,01	102,01	0,00%
38	157,06	157,01	3,18%	101,99	102,01	1,96%
39	156,95	157,02	4,46%	102,05	102,01	3,92%
40	156,93	157	4,46%	101,93	101,98	4,90%
41	156,98	157,01	1,91%	102,00	101,96	3,92%
42	156,98	156,99	0,64%	102,00	101,95	4,90%
43	157,02	156,98	2,55%	101,99	101,96	2,94%
44	156,92	157,02	6,37%	102,01	101,99	1,96%
45	157,02	157,01	0,64%	101,99	101,95	3,92%
46	156,92	156,98	3,82%	102,06	102,02	3,92%
47	157,01	156,99	1,27%	101,95	102,00	4,90%
48	157,00	157	0,00%	101,97	101,99	1,96%
49	156,98	157,02	2,55%	101,93	102,00	6,86%
50	156,96	156,98	1,27%	102,07	101,98	8,83%
51	156,95	157,01	3,82%	101,97	101,99	1,96%
52	157,05	157	3,18%	102,01	101,99	1,96%
53	157,06	157	3,82%	102,06	101,99	6,86%
54	157,04	156,99	3,18%	102,02	102,00	1,96%
55	157,00	156,98	1,27%	102,05	101,96	8,83%
56	156,96	157,01	3,18%	102,06	101,97	8,83%
57	157,05	157,02	1,91%	102,01	102,02	0,98%
58	156,97	157	1,91%	101,94	102,02	7,84%
59	157,05	157	3,18%	101,98	101,96	1,96%
60	157,04	157	2,55%	101,94	102,01	6,86%
61	156,92	156,99	4,46%	101,93	102,01	7,84%
62	156,94	157,02	5,09%	102,04	102,01	2,94%
63	157,05	156,98	4,46%	102,07	101,97	9,81%
64	157,03	156,99	2,55%	102,01	101,95	5,89%
65	157,06	156,99	4,46%	101,97	101,97	0,00%
66	157,04	157,01	1,91%	102,05	101,96	8,83%
67	156,93	157	4,46%	102,07	101,99	7,84%
68	157,00	156,99	0,64%	101,99	101,98	0,98%

69	156,99	157,01	1,27%	101,98	101,95	2,94%
70	156,93	157	4,46%	102,00	101,96	3,92%
71	157,02	157	1,27%	102,03	102,00	2,94%
72	157,03	157,02	0,64%	101,98	101,98	0,00%
73	157,04	156,98	3,82%	101,93	102,02	8,82%
74	156,95	157	3,18%	102,01	101,95	5,89%
75	156,96	156,99	1,91%	102,05	101,99	5,88%
76	156,93	157	4,46%	101,93	102,01	7,84%
77	156,98	157	1,27%	102,04	102,02	1,96%
78	156,94	157,02	5,09%	102,06	102,02	3,92%
79	156,94	156,98	2,55%	102,01	102,00	0,98%
80	157,02	157,02	0,00%	102,07	102,01	5,88%
81	157,06	157	3,82%	102,06	101,95	10,79%
82	157,03	157,02	0,64%	102,00	102,00	0,00%
83	156,93	156,99	3,82%	102,04	101,99	4,90%
84	156,98	157	1,27%	101,97	101,97	0,00%
85	156,94	156,98	2,55%	101,93	101,97	3,92%
86	157,00	156,99	0,64%	101,99	102,01	1,96%
87	156,95	157,01	3,82%	101,98	101,98	0,00%
88	156,97	156,99	1,27%	102,01	102,02	0,98%
89	156,95	156,98	1,91%	102,02	102,01	0,98%
90	157,02	156,99	1,91%	101,97	102,02	4,90%
91	156,99	157	0,64%	101,99	102,02	2,94%
92	156,96	157	2,55%	102,02	101,95	6,87%
93	157,03	156,98	3,19%	101,93	102,01	7,84%
94	157,06	157	3,82%	102,04	101,96	7,85%
95	157,06	156,98	5,10%	102,06	102,02	3,92%
96	157,00	157	0,00%	102,07	101,95	11,77%
97	156,96	157,01	3,18%	102,03	102,02	0,98%
98	156,93	156,99	3,82%	102,03	102,01	1,96%
99	156,96	157,02	3,82%	101,98	102,01	2,94%
100	156,97	157,02	3,18%	101,99	101,99	0,00%