

Proyecto de Grado

Sábado 9 de diciembre de 2023

Metrología Automatizada: Caracterización de Elementos Mecánicos con Visión Artificial y Redes Neuronales

Autor: DANIEL ANDRES BUITRAGO VELASCO

Código: 202014164

d.buitragov@uniandes.edu.co

Asesor:

EDGAR ALEJANDRO MARAÑÓN LEON

UNIVERSIDAD DE LOS ANDES

DEPARTAMENTO DE INGENIERÍA MECÁNICA

FACULTAD DE INGENIERÍA

Bogotá D.C., Colombia

RESUMEN

Este proyecto propone un programa enfocado en la caracterización automática de eslabones binarios, elementos fundamentales en numerosas estructuras mecánicas. Su esencia radica en el uso de una simple fotografía del elemento, a partir de la cual se derivan sus dimensiones más relevantes y se identifican sus modos de falla predominantes. Este proceso de evaluación se lleva a cabo mediante la integración de técnicas de visión por computadora e inteligencia artificial. La combinación de estas tecnologías no solo acelera la identificación de potenciales fallas, sino que también proporciona datos cruciales para la toma de decisiones informadas. Este proyecto se encuentra en la intersección entre la tecnología digital y el análisis de componentes mecánicos, ofreciendo una alternativa automática a los métodos de metrología tradicionales, siendo accesible para todo público y con un nivel de precisión aceptable.

ABSTRACT

This project proposes a program focused on the automatic characterization of binary links, fundamental elements in numerous mechanical structures. Its particularity lies in the use of a simple photograph of the element, from which its most relevant dimensions are derived and its predominant failure modes are identified. This evaluation process is carried out by integrating computer vision and artificial intelligence techniques. The combination of these technologies not only accelerates the identification of potential failures, but also provides crucial data for informed decision making. This project is located at the intersection between digital technology and the analysis of mechanical components, offering an automatic alternative to traditional metrology methods, being accessible to the general public and with an acceptable level of precision.

Palabras clave: Eslabones binarios, fotografía, redes neuronales, visión por computadora.

NOMENCLATURA

$Dist_c$	Distancia entre centros: Distancia entre los centros de los agujeros de los extremos del eslabón.	cm
D_0	Diámetro interior: Diámetro de los nodos del eslabón	cm
W	Ancho del eslabón	cm
t	Espesor del eslabón	cm
D_p	Diámetro del pasador	cm
D_L	Distancia al borde	cm
D_c	Distancia del centro del pasador al borde	cm
F_{app}	Fuerza aplicada al eslabón	kN
S_{ut}	Resistencia última a la tracción del material del eslabón	MPa
S_{bru}	Resistencia al rodamiento	MPa
S_{su}	Resistencia última a cizalladura del material del eslabón	MPa
E	Módulo de elasticidad	GPa
C_n	Centroide de la sección n del eslabón	Coordenadas en pixeles
A_n	Área de la sección n del eslabón	Pixeles cuadrados
FS_m	Factor de seguridad por modo de falla m	
F_{cm}	Fuerza crítica para el modo de falla m	kN

Tabla 1. Nomenclatura

Tabla de contenido

RESUMEN	2
NOMENCLATURA	3
OBJETIVOS	6
Objetivo general	6
Objetivos específicos	6
INTRODUCCIÓN	7
MARCO TEORICO	9
Elección de objeto de estudio	9
Características del eslabón binario	9
Procedimiento de medición	11
Selección de referencia	11
Momentos geométricos e identificación de medidas	12
Estimación de dimensiones	13
Modos de falla del eslabón binario	15
Fallo de tensión en la sección neta	17
Fallo por corte a lo largo de dos planos	18
Falla de cizallamiento	20
Falla por hendidura	21
Herramientas y tecnologías	21
Funcionalidad de una red neuronal	22
METODOLOGÍA	24
Captura de la imagen del eslabón	26
Arreglos iniciales a la imagen	27
Identificación del contorno de la hoja	27
Transformación de perspectiva	28
Arreglos de imagen y binarización	29
Obtención de los contornos del eslabón	30
Encontrar las medidas características del eslabón	31
Red neuronal RN1	31
Estimar modos de falla	33

Datos extra requeridos.....	33
Red neuronal RN2	34
Visualizar modos de falla.....	36
RESULTADOS	38
Dimensiones características	38
Modos de falla	40
Costo computacional.....	42
DISCUSIÓN	43
CONCLUSIÓN	46
BIBLIOGRAFÍA	47
ANEXOS	49

OBJETIVOS

Objetivo general

Desarrollar un programa basado en métodos de visión por computadora y redes neuronales para la caracterización precisa de eslabones binarios.

Objetivos específicos

1. Identificar el eslabón binario proporcionado independientemente de su forma o disposición en la imagen.
2. Identificar las dimensiones clave del elemento proporcionado utilizando técnicas de visión por computadora.
3. Verificar la precisión del programa frente a los métodos de metrología convencionales.
4. Caracterizar los modos de falla predominantes del elemento proporcionado a través de redes neuronales.

INTRODUCCIÓN

La metrología es la disciplina encargada del estudio, desarrollo y aplicación de técnicas de medición. Su importancia radica en garantizar la precisión y calibración de componentes, siendo esencial tanto en la optimización de procesos como en la verificación de estándares post-producción (SYBRIDGE Technologies, 2020). Esta ciencia resulta crucial en la ingeniería mecánica, cuyos estándares industriales en continua evolución exigen una precisión cada vez más elevada, motivando así la creación de instrumentos sofisticados, frecuentemente costosos (International Journal of Metrology and Quality Engineering (IJMQE), 2019). Este panorama se complementa con la creciente automatización de procesos, creando la necesidad de explorar alternativas a los instrumentos de metrología convencionales, más accesibles económicamente y capaces de garantizar niveles de precisión aceptables.

Es en este contexto surge el propósito de este proyecto: implementar un programa capaz de servir como alternativa automatizada a la metrología convencional. Para lograrlo, se opta por la utilización de la biblioteca OpenCV de Python, elegida sobre otros algoritmos como los ofrecidos por Matlab debido a la versatilidad y facilidad de uso que ofrece Python. (Paradigma, 2017). Esta elección permite emplear métodos de visión artificial mediante técnicas como la identificación de contornos, cálculo de momentos de imagen y operaciones morfológicas, entre otras.

Para el desarrollo del programa, se emplea un enfoque centrado en eslabones binarios, elementos simples y presentes en aplicaciones de transmisión de potencia y movimientos articulados, como las cadenas de transmisión. A partir de estos eslabones, se identifican las condiciones de uso necesarias para una aplicación como la aquí presentada, incluyendo sus requerimientos, limitaciones y alcance para piezas de mayor complejidad.

También se explora la sinergia entre visión e inteligencia artificial, un campo en constante desarrollo con el cual es posible el reconocimiento de figuras y la extracción de patrones a imágenes (Garcia Garcia , 2013), cuya aplicación dentro de la ingeniería

mecánica no ha sido explorada exhaustivamente. Esto permite integrar un componente adicional a este proyecto en comparación con la metrología convencional, al ofrecer información extra sobre las características de las piezas como sus modos de falla y condiciones operativas. Características que serán halladas mediante el empleo de redes neuronales, una rama específica de la inteligencia artificial, cuya creación es posible a partir de la biblioteca de TensorFlow y Keras de Python. Para esto se plantea la creación de una interfaz intuitiva utilizando Tkinter, otra librería de Python. Lo cual permite a los usuarios cargar imágenes de eslabones binarios en el programa para el análisis. Como resultado, la aplicación identifica tanto las dimensiones como las posibles fallas de estos eslabones, independientemente de su disposición o forma en la imagen proporcionada.

Este proyecto no solo pretende representar un avance significativo en la metrología, sino que sienta las bases para posibles aplicaciones de técnicas de metrología en piezas mecánicas diferentes a un eslabón binario. La metodología y las herramientas desarrolladas aquí tienen el potencial de adaptarse y expandirse hacia la medición y caracterización de componentes de mayor complejidad dimensional, ofreciendo un programa inicial para la exploración de aplicaciones más amplias en la ingeniería mecánica.

MARCO TEORICO

En este marco teórico se establecen las bases conceptuales necesarias para el desarrollo del proyecto. Se abordan aspectos clave de los objetos de estudio, como el porqué de su elección, su definición, sus principales características y modos de falla predominantes.

Elección de objeto de estudio

Para iniciar el desarrollo de un proyecto destinado a la caracterización de objetos mecánicos, es prudente la elección de elementos simples dimensionalmente. Esto se realiza con el objetivo de sentar las bases del proyecto, comprender sus características y establecer las dimensiones del problema.

Entre las opciones consideradas, se evaluaron diversos elementos mecánicos, entre ellos, distintos tipos de eslabones, tornillos, tuercas, chavetas, bielas, vástagos, ejes, rodamientos o inclusive engranajes. No obstante, se optó por enfocar el análisis en los eslabones binarios, debido a su baja complejidad dimensional, lo que facilita enormemente su cuantificación en comparación con cualquier otro elemento mecánico mencionado anteriormente. Esto lo convierte en objeto de estudio idóneo para sentar los cimientos de este proyecto.

Características del eslabón binario

Un eslabón binario es un cuerpo rígido que posee dos nodos, siendo éstos los puntos de unión con otros eslabones facilitando la flexibilidad y el movimiento articulado entre ellos (Sánchez, 2008). Esta pieza es usada en aplicaciones de transmisión de potencia y movimiento articulado, como es el caso de las cadenas de transmisión. Pueden encontrarse en diversos tamaños, desde pequeños conectores hasta piezas de gran dimensión, que

aunque presentan una estructura simple, son esenciales en una diversidad de mecanismos y funcionan como elementos sometidos generalmente a tensión. (HWANG & CHEN, 1995)



Ilustración 1. Eslabones binarios de una cadena de bicicleta enlazados mediante pernos

En este proyecto, enfocaremos la atención en la simplicidad dimensional de los eslabones binarios, ya que su caracterización puede ser realizada mediante tres medidas fundamentales: la distancia entre sus centros ($Dist_C$), el diámetro interno (D_0) propio de sus nodos y el ancho del eslabón (W).

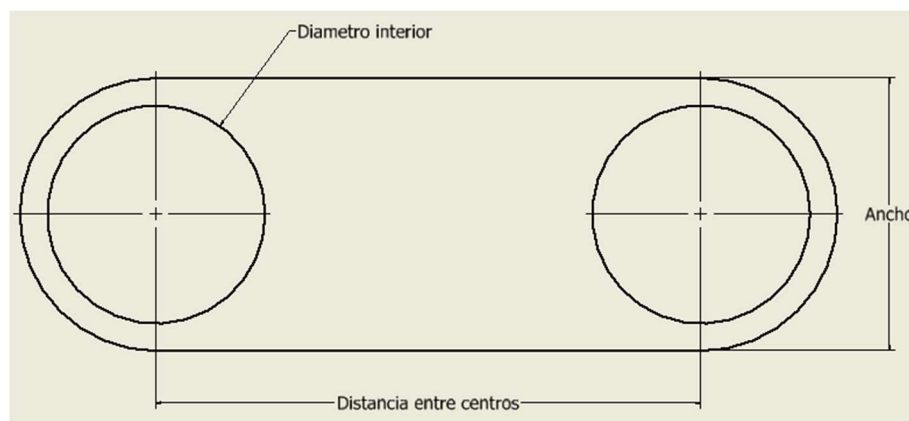


Ilustración 2. Caracterización del eslabón binario

caracterización es posible siempre y cuando los nodos estén centrados respecto a los semicírculos extremos del eslabón y que los círculos de los nodos tengan el mismo diámetro, como se aprecia en la ilustración 2. Estas medidas son fácilmente identificables a simple vista, por tanto, podrán ser cuantificadas mediante el análisis de una simple imagen.

Procedimiento de medición

Implementar un programa para medir este tipo de elementos se fundamenta en ofrecer una solución automatizada en el proceso de caracterización de componentes mecánicos. Esto conlleva algunas ventajas como la agilización del análisis, la estandarización del proceso de evaluación e inclusive una posible mejora en la eficiencia y exactitud de la medición.

Selección de referencia

La medición de un eslabón binario a partir de una imagen requiere de una referencia. En la metrología tradicional, se utilizan de referencia reglas o calibradores, pero en un entorno digital, es conveniente idear un equivalente fácilmente reconocible en una imagen como apoyo. Considerando varias opciones para lograr este propósito, se evaluaron alternativas como el uso de monedas u otros objetos dentro de las imágenes. Sin embargo, esto conlleva a varias complicaciones, ya que las monedas no presentan una estandarización global en términos de tamaño y espesor, lo que dificulta su aplicación generalizada. Usar una regla dentro de la imagen también generaría problemas, ya que su identificación, la referencia utilizada (ya sea centímetros o pulgadas), e incluso la variabilidad de sus longitudes entre diferentes modelos, implica una complejidad adicional para el programa.

Otra opción considerada fue el uso de una malla o cuadrícula como marco de referencia. Aunque esto puede ser estandarizado, la necesidad de disponer de una malla para cada medición no es práctica en la mayoría de los casos. Lo que conlleva a pensar en la posibilidad de usar hojas de papel cuadriculado, pero su falta de uniformidad en la disposición de los cuadros y la posible dificultad de identificar un cuadro específico como referencia presentan obstáculos similares a los encontrados con la regla.

Dada la necesidad de encontrar una referencia ampliamente disponible y estandarizada, se optó por el uso de una hoja tamaño carta estándar (A4) como marco de referencia. Estas hojas miden 21.59 cm de ancho y 27.94 cm de largo y son completamente

monocromáticas, sin patrones internos disuasorios para facilitar la identificación clara de sus contornos y minimizar la complejidad del proceso.

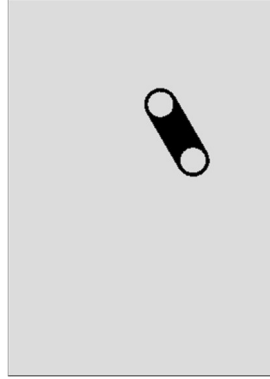


Ilustración 3. Eslabón binario dispuesto sobre una hoja tamaño carta

En este orden de ideas, se coloca un eslabón dispuesto al azar dentro de la hoja y el programa estima sus dimensiones a partir de esta.

Momentos geométricos e identificación de medidas

Para identificar las medidas características del eslabón se usan momentos geométricos o momentos de Hu, funciones invariantes a transformaciones geométricas como translación, escalado o rotación, obteniendo una herramienta valiosa para reconocer una imagen independientemente de su orientación en un eje de coordenadas. Por ejemplo, la imagen de un eslabón como se muestra en la ilustración 3, será muy distinta si este está centrado o rotado, no obstante, la figura, en sí, es la misma (Universidad de Sevilla, 2015).

Para una imagen discreta, el momento de orden i, j está definido por las coordenadas de sus píxeles, expresadas como números enteros. Así, los momentos geométricos reflejan la intensidad de los píxeles $I(x, y)$ y se calculan de la siguiente manera:

$$M_{ij} = \sum_x \sum_y x^i y^j I(x, y)$$

Para el caso en el que se trate con imágenes binarias, el valor de $I(x, y)$ será 0 o 1, lo cual simplifica la formula anterior, considerando dominio de coordenadas (x, y) donde $I(x, y) = 1$:

$$M_{i,j} = \sum_x \sum_y x^i y^j$$

Ecuación 1.1. Momentos geométricos para una imagen binarizada

Dos propiedades básicas de la imagen se derivan de los momentos espaciales de orden 0 y 1: (Universidad de Sevilla, 2015)

- Área de una imagen binaria: M_{00}
- Centroide $\{\bar{x}, \bar{y}\} = \left\{ \frac{M_{10}}{M_{00}}, \frac{M_{01}}{M_{00}} \right\}$

Ecuaciones 1.2. Cálculos de área y centroide a partir de los momentos geométricos.

Para el desarrollo del proyecto, bastará con estas 2 variables para caracterizar completamente el eslabón binario puesto que de ellas se podrá inferir la distancia entre centros, el diámetro interno y el ancho del eslabón.

Estimación de dimensiones

Se pueden obtener 3 momentos de imagen propios de diferentes secciones específicas para un mismo eslabón binario: los 2 agujeros de sus pasadores (Secciones 1 y 2) y el eslabón en sí (Sección 3). Esto permite obtener tanto los centroides como el área para cada sección.

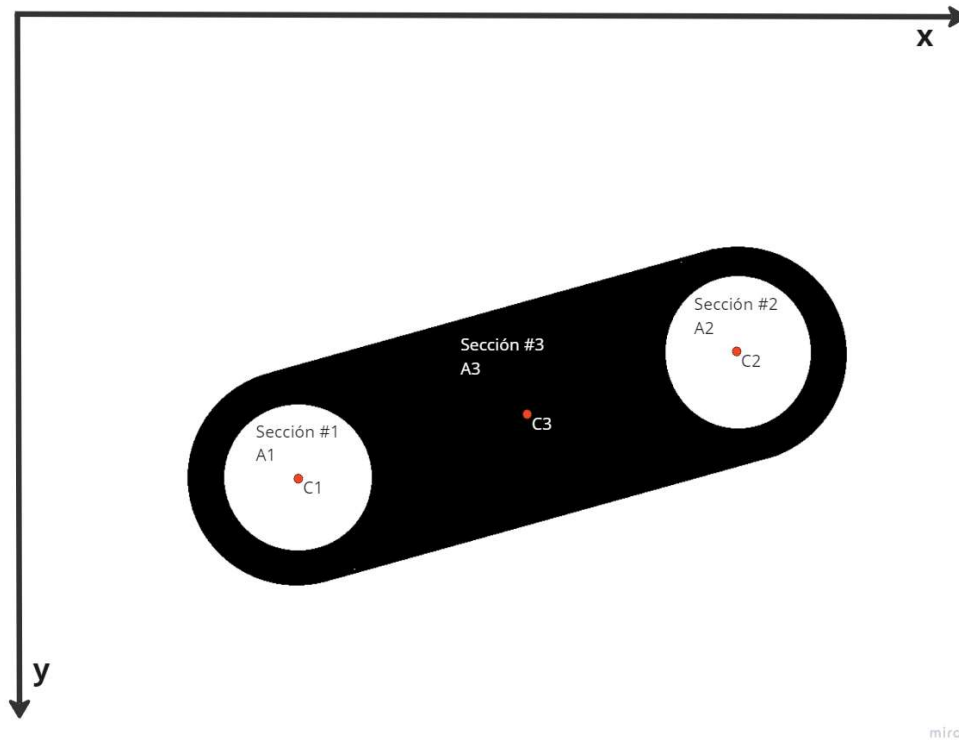


Ilustración 4. Secciones de un eslabón binario.

En la ilustración anterior se muestra un eslabón binario sobre el eje de coordenadas de OpenCV, cuyo centro de origen está en la esquina superior izquierda. Los puntos en rojo identifican los centroides de cada sección mostrados como $C_1 = (x_1, y_1)$, $C_2 = (x_2, y_2)$ y $C_3 = (x_3, y_3)$ cuyas ubicaciones se definen por coordenadas en píxeles. Cada sección tiene su área respectiva, representada como A_1, A_2 y A_3 en píxeles cuadradas.

De esta manera, la distancia entre centros ($Dist_C$) será la distancia entre los centroides C_1 y C_2 de ambos agujeros:

$$Dist_C = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

Ecuación 2. Cálculo para la distancia entre centros

Con las áreas, se podrán derivar tanto el diámetro interior como el ancho, considerando que el área de los agujeros está dada por:

$$A_{1,2} = \frac{\pi D_0^2}{4}$$

Ecuación 3. Áreas de los agujeros del eslabón

El área total del eslabón binario se puede estimar como la suma del área de un rectángulo de ancho W y largo $Dist_C$ más la de un círculo de diámetro equivalente a W , restando el área de los agujeros:

$$A_3 = \pi W^2 + W Dist_C - \frac{\pi D_0^2}{2}$$

Ecuación 4. Área del eslabón binario

Mediante estas fórmulas se puede estimar las dimensiones D_0 y W despejando. Sin embargo, en lugar de calcularlas directamente, se utilizarán como base para entrenar una red neuronal (RN1). Esta red neuronal será capaz de estimar D_0 y W , sin requerir las fórmulas explícitamente. Esto será explicado detalladamente en la sección de Herramientas y Tecnologías.

Modos de falla del eslabón binario

El análisis a un eslabón binario es engañosamente complejo ya que existen varios modos de falla interrelacionados. Dichos modos están asociados a distintas áreas de la orejeta del eslabón, cada uno representando una posible fuente de deterioro o colapso estructural. Este análisis es esencial para cuantificar dichos fallos, sin embargo, para este proyecto se empleará un enfoque simplificado con base en suposiciones sobre la naturaleza de la falla, lo que permite una aproximación adecuada para los propósitos del análisis.

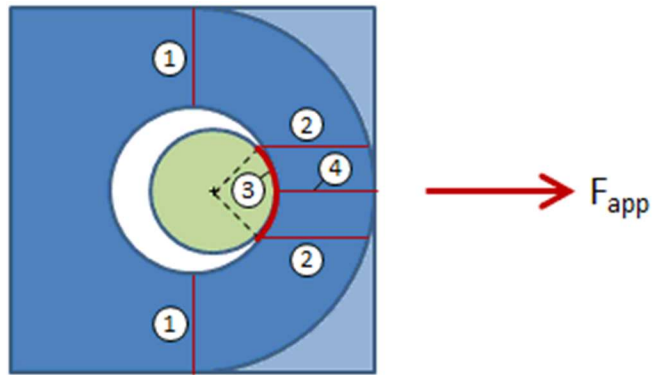


Ilustración 5. Modos de falla del extremo del eslabón. (MechaniCalc, 2014)

En la ilustración anterior se muestran los modos de falla comunes representados en rojo sobre la orejeta del eslabón binario en azul y el pasador en verde. Los modos de falla del eslabón se enumeran a continuación. Los números corresponden con las secciones etiquetadas de la ilustración 3:

1. Falla por tensión en la sección neta
2. Falla por corte a lo largo de dos planos
3. Falla de cizallamiento
4. Falla por hendidura

Para cada uno de los modos de falla, se calculará un factor de seguridad. Siempre que el factor de seguridad sea aceptable para un modo de falla determinado ($n_{Falla} > 1$), se considerará que el elemento no falla por ese modo. Las dimensiones necesarias para realizar este análisis se detallan en la siguiente figura:

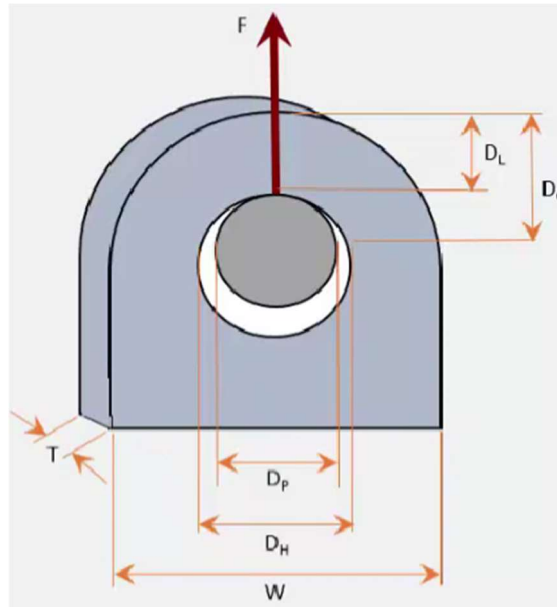


Ilustración 6. Dimensiones para un análisis simplificado del eslabón. (Engineering Solved, 2016)

Las dimensiones en la ilustración 5 son visualizadas en la tabla 1 de nomenclatura.

Tanto D_L como D_C son estimados a partir de la caracterización mostrada en la ilustración 2, pero t y D_P tendrán que ser agregados por el usuario una vez caracterizado el eslabón. Además de esto, las propiedades del material del eslabón y pasador también deberán ser agregados por aparte, al igual de la fuerza a la cual estará sometido, pues a partir de una imagen es imposible inferirlos.

A continuación, se mostrarán los métodos de cálculo para cada uno de los modos de falla identificados:

1. Fallo de tensión en la sección neta

Ocurre cuando la tensión aplicada supera la capacidad de carga del material en la sección más delgada del eslabón. (MechaniCalc, 2014)

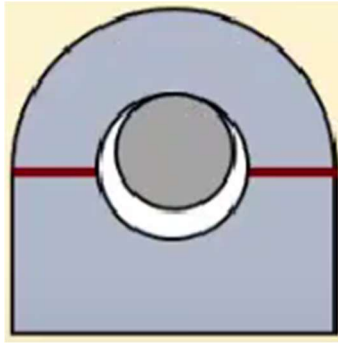


Ilustración 7. Fallo de tensión en la sección neta. (Engineering Solved, 2016)

La determinación del área de la sección neta se calcula mediante la fórmula:

$$A_t = (w - D_0)t$$

La carga última P_{tu} es aquella que provocaría la falla por tensión a lo largo de la sección y se determina mediante la fórmula:

$$F_{C_1} = S_{tu} * A_t$$

Ecuación 5.1. Fuerza critica para el modo de falla 1

Donde S_{tu} representa la resistencia última a la tracción del material del eslabón. Esta ecuación asume una distribución uniforme de la tensión de tracción sobre la sección. Sin embargo, en realidad, existirá una concentración de esfuerzos debido al flujo de tensión alrededor del agujero.

El factor de seguridad se obtiene mediante la relación:

$$FS_1 = \frac{F_{C_1}}{F_{app}}$$

Ecuación 5.2. Factor de seguridad para el modo de falla 1

Donde F_{app} es la carga aplicada sobre el eslabón.

2. Fallo por corte a lo largo de dos planos

Representa una falla debido a la ruptura del eslabón en dos áreas distintas sometidas a fuerzas de corte. (MechaniCalc, 2014)

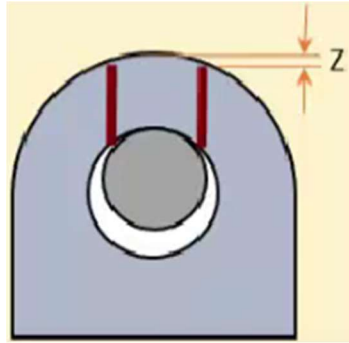


Ilustración 8. Fallo por corte a lo largo de dos planos (Engineering Solved, 2016)

El área total de los planos de corte se calcula como:

$$A_s = 2 * L_{sp} * t$$

Donde L_{sp} es la longitud del plano de corte. Para su cálculo se suele considerar una línea de 40 grados que se extiende desde el centro del pasador de corte. En el punto donde dicha línea cruza con el orificio del pasador, se extiende el plano de corte horizontalmente hasta el borde exterior de la orejeta. L_{sp} es calculado como:

$$L_{sp} = D_L + \frac{D_p}{2} (1 - \cos 40^\circ) - Z$$

Donde Z es la pérdida de longitud del plano de corte debido a la curvatura en el extremo del eslabón, calculado como:

$$Z = \frac{W}{2} - \sqrt{\left(\frac{W}{2}\right)^2 - \left(\frac{D_p}{2} \sin 40^\circ\right)^2}$$

La carga última P_{su} es aquella que resultaría en el desgarro por corte a lo largo de los dos planos, y se determina mediante la fórmula:

$$F_{C_2} = S_{su} * A_s$$

Ecuación 6.1. Fuerza critica para el modo de falla 2

Aquí, S_{su} es la resistencia última a la cizalladura del material del eslabón.

El factor de seguridad se calcula como:

$$FS_2 = \frac{F_{c_2}}{F_{app}}$$

Ecuación 6.2. Factor de seguridad para el modo de falla 2

3. Falla de cizallamiento

Esta falla ocurre entre la superficie del pasador y la superficie interior del orificio del eslabón y provoca un aplastamiento del eslabón en los puntos de contacto. (MechaniCalc, 2014)

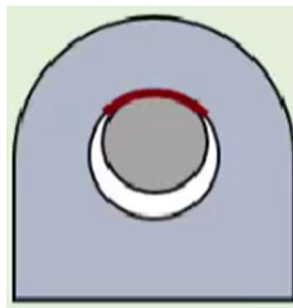


Ilustración 9. Falla de cizallamiento. (Engineering Solved, 2016)

El área de soporte se calcula como:

$$A_{br} = D_p * t$$

La carga última P_{bru} es aquella que causaría la falla por rodamiento y se determina como:

$$F_{C_3} = S_{bru} * A_{br}$$

Ecuación 7.1. Fuerza crítica para el modo de falla 3

Donde S_{bru} es el mínimo entre la resistencia última por carga del material del eslabón, la resistencia última por carga del material del pasador y 1.5 veces la resistencia máxima de tensión del eslabón:

$$S_{bru} = \min(S_{ub_{eslabon}}, S_{ub_{pasador}}, 1.5 * S_{ut})$$

El factor de seguridad es entonces:

$$FS_3 = \frac{F_{C_3}}{F_{app}}$$

Ecuación 7.2. Factor de seguridad para el modo de falla 3

4. Falla por hendidura

Esta falla implica la ruptura a lo largo de una única área del eslabón. (Engineering Solved, 2016)

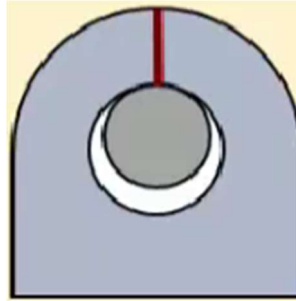


Ilustración 8. Falla por hendidura

El área de tensión se calcula como:

$$A_t = D_L * t$$

La carga última P_{bru} es aquella que causaría la falla por tensión en el plano único y se calcula con la fórmula:

$$F_{C_4} = S_{tu} * A_t$$

Ecuación 8.1. Fuerza crítica para el modo de falla 4

Donde S_{tu} representa la resistencia última a la tracción del material del eslabón nuevamente.

El factor de seguridad se calcula como:

$$FS_4 = \frac{F_{C_4}}{F_{app}}$$

Ecuación 8.2. Factor de seguridad para el modo de falla 4

Herramientas y tecnologías

Se usó el lenguaje de programación Python debido a su versatilidad y concordancia con los objetivos planteados. Esta elección se fundamenta en la presencia de bibliotecas como OpenCV y TensorFlow, elementos clave en el desarrollo del proyecto.

OpenCV es la librería principal con la cual es posible el procesamiento de imágenes. Permite la manipulación, análisis y extracción de información de las imágenes entregadas para su posterior evaluación (OpenCV, 2000). Por otro lado, TensorFlow, también biblioteca de código abierto desarrollada por Google, es empleada en el desarrollo de redes neuronales (TensorFlow, 2015). Estas redes son usadas tanto en la identificación de los modos de falla de los eslabones como en la caracterización de estos a partir de imágenes previamente procesadas.

Esta integración permite aprovechar la sinergia entre las capacidades de procesamiento de imágenes de OpenCV y las funcionalidades de aprendizaje profundo de TensorFlow. De esta forma es posible enriquecer el proyecto con un enfoque más atractivo, amplía sus posibilidades y añade un nivel adicional de desafío y profundidad al proyecto.

Funcionalidad de una red neuronal

Aunque se han presentado métodos para calcular tanto las dimensiones características (véase ecuaciones 3 y 4) como los modos de falla (véase ecuaciones 5, 6, 7 y 8), no se utilizarán directamente en el proceso de estimación. En su lugar, se utilizarán como base para entrenar redes neuronales.

Las redes neuronales son un componente esencial en la creación de inteligencia artificial, una disciplina que, según (McCarthy, 2007): “Es la ciencia e ingeniería de hacer máquinas inteligentes, especialmente programas informáticos inteligentes.” Esto con el fin de realizar tareas complejas, entender comportamientos o patrones y establecer funciones, similar a como lo haría un ser humano. En esencia, estos sistemas buscan “pensar” o actuar de manera similar a los humanos, como destaca IBM. (IBM, 2012). Para esto se aplican redes neuronales, complejos algoritmos que emulan el aprendizaje controlado del cerebro humano. Son compuestas por múltiples capas, incluyendo una capa de entrada, una o más capas ocultas y una capa de salida. Cada capa contiene nodos que simulan la funcionalidad de las neuronas humanas. Cada nodo, o neurona artificial, tiene un peso y umbral asociado, cuya combinación mediante una suma ponderada de los pesos de los nodos de la capa

inmediatamente anterior, logran, en conjunto, resultados asombrosos. Una red neuronal tiene la capacidad de ajustar dichos valores de peso y umbral asociado a cada neurona en base en un set de entrenamiento, este ajuste se conoce popularmente como el “aprendizaje de la red neuronal” (IBM, 2014).

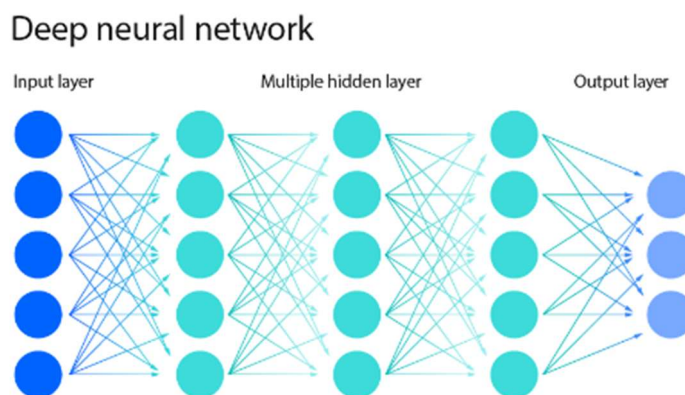


Ilustración 10. Estructura de una red neuronal sacada de IBM

Para este proyecto, se implementarán dos redes neuronales. La primera (*RN1*) se entrenará para aprender a obtener los valores de D_0 y W a partir de las áreas A_1 , A_2 y A_3 . Este proceso, un humano lo resolvería intuyendo las ecuaciones 3 y 4 y luego despajando, pero ¿Y una maquina? Pues, también podrá llegar hacer este proceso con base en un set de entrenamiento basado en estas ecuaciones. La segunda red (*RN2*) aprenderá a estimar las fuerzas críticas por modo de falla. Esta tarea, que podría resolverse manualmente intuyendo las ecuaciones 5, 6, 7 y 8, será posible que la haga el programa enteramente mediante un set de entrenamiento basado en dichas ecuaciones.

La estructura de cada una depende de la complejidad y naturaleza del problema, pero, lamentablemente no hay una manera de determinar el numero de neuronas y capas de una red neuronal con solo la descripción del problema (Luca, 2022). No obstante, con algunos parámetros medibles durante el proceso de aprendizaje es posible aproximarse a una combinación que mejor se adapte el problema. Hay algunas formas de optimización de la estructura con base en dichos parámetros que están enumeradas y condensadas en el

artículo de (Nowakowski, Dorogyy, & Doroga-Ivaniuk, 2017) al respecto. Aquí se propone una sucesiva adición de capas ocultas, de salida, formas de extraer información de capas internas, y demás métodos fácilmente programables desde TensorFlow que serán aplicadas.

Esta aproximación es producto de una inmensa curiosidad sobre el funcionamiento de estas extraordinarios herramientas, que, en un futuro, muy probablemente cambien para bien nuestra sociedad.

METODOLOGÍA

La metodología propuesta se centra en un proceso secuencial que abarca desde la captura de la imagen del eslabón binario hasta la evaluación de sus potenciales modos de falla. Esto se realiza mediante un paso a paso en el que el usuario interactúa con el programa que se llamó como: “**Analizador de Figuras**”.

El proceso consta de 8 fases distintas:

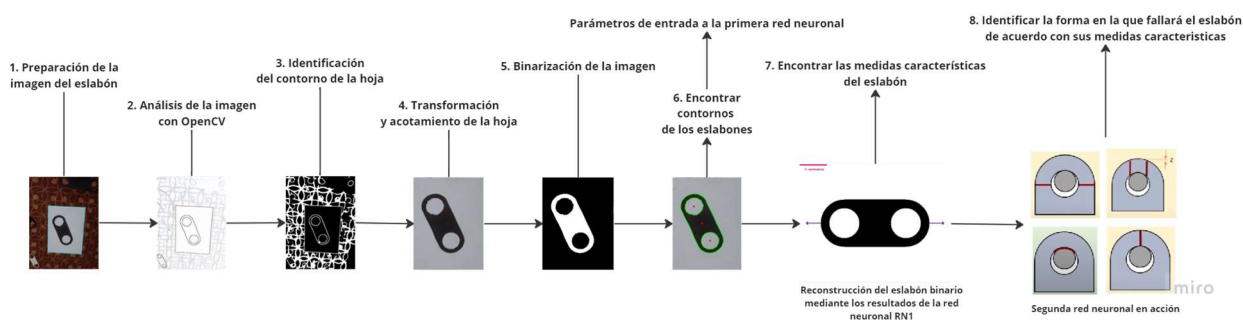


Ilustración 11. Metodología paso a paso

La realización de este algoritmo se apoyó en el uso de una interfaz hecha mediante Tkinter, lo que posibilita una interacción dinámica con el usuario para recibir información adicional y presentar resultados. Esto implica una fácil notificación de errores a través de la interfaz, permitiendo la posibilidad de solicitar al usuario que se agregue, cambie o elimine información sin interrumpir la ejecución del programa.

La explicación de cada paso se hace con base en las siguientes tres imágenes:



Ilustración 12.1. Primera figura de análisis

Figura 1: (23)

- Imagen con sombras.
- Hoja de papel rasgada ligeramente e irregular.
- Eslabón con desperfectos en sus contornos.



Ilustración 12.2. Segunda figura de análisis

Figura 2: (9)

- Hoja sobre superficie irregular.
- Foto tomada lejos de la referencia.
- Foto desenfocada.
- Eslabón con bordes irregulares
- Eslabón no centrado en la hoja.



Ilustración 12.3. Tercera figura de análisis

Figura 3: (46)

- Hoja negra y eslabón blanco.
- Foto con flash.
- Eslabón pequeño.

Todas estas imágenes fueron diseñadas con cartulina, lo que implica que su creación fue un trabajo de manualidades que lleva, en la mayoría de los casos, a desperfectos en sus cortes y dimensiones. Mediante estas imágenes será posible entender el funcionamiento del programa, sus limitaciones, aciertos y oportunidades de mejora a medida que avanzamos en la metodología.

1. Captura de la imagen del eslabón

El desarrollo del programa inicia con la captura de la imagen que contiene el eslabón binario a analizar. La interfaz da la bienvenida y proporciona las instrucciones necesarias para su proceder. En este sentido, se recomienda al usuario suministrar una imagen nítida, exenta de brillos, sombras u otros elementos que puedan interferir con la identificación del eslabón.

En la imagen deberá estar una hoja tamaño carta que contenga el eslabón de estudio en su interior tal y como se explicó en la sección “*obtención de dimensiones del eslabón a partir de una imagen*” del marco teórico. Es crucial que la imagen contenga ambos elementos, la hoja y el eslabón, con un claro contraste entre ellos para permitir su identificación.

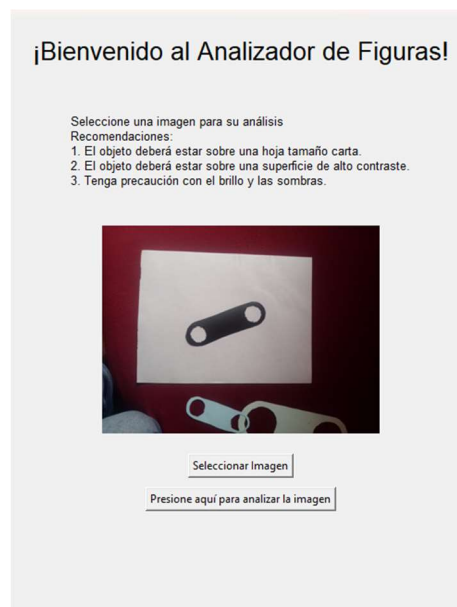


Ilustración 13. Primera ventana de la interfaz para el ingreso de la foto a analizar.

Esta primera ventana de la interfaz denominada: “Analizador de eslabones binarios” busca garantizar que la imagen proporcionada cumpla con los criterios para realizar una evaluación precisa del eslabón, asegurando un análisis efectivo en las etapas posteriores del programa.

2. Arreglos iniciales a la imagen

El programa empieza resaltando los bordes de la hoja tamaño carta en la imagen original mediante herramientas de OpenCV. Este proceso implica una serie de operaciones morfológicas, empezando con la dilatación de la imagen para amplificar los bordes. Posteriormente, se aplica un filtro de desenfoque gaussiano con un kernel de 5x5 para suavizar la imagen y eliminar posibles imperfecciones o ruidos en los bordes. Finalmente, se normaliza para realzar aún más los contornos.

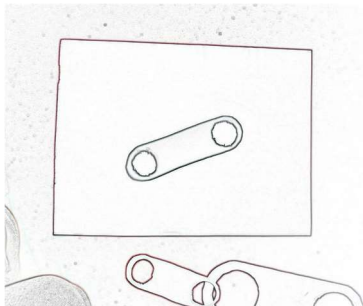


Ilustración 14.1. Fig 1

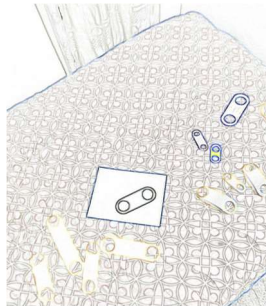


Ilustración 14.2. Fig 2



Ilustración 14.3. Fig 3

3. Identificación del contorno de la hoja

Una vez resaltado el borde de la hoja, se utiliza la función *findContours* de OpenCV para detectar los contornos presentes en la imagen. Esta función detecta los contornos basándose en las variaciones de intensidad entre píxeles adyacentes (Geeks For Geeks, 2023). Aquellos contornos que superen cierto tamaño, se someten a un análisis



Ilustración 15.1. Fig 1

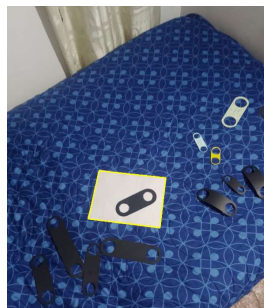


Ilustración 15.2. Fig 2



Ilustración 15.3. Fig 3

adicional para identificar el contorno que representa los bordes de la hoja. Para esto se utiliza la función *approxPolyDP*, la cual aproxima la forma del contorno a un polígono con un número mínimo de vértices (Geeks For Geeks, 2023). En este caso, aquel contorno que presente cuatro vértices, será entendido como el de la hoja tamaño carta.

Cabe destacar que la aproximación de una hoja a un rectángulo perfecto puede presentar desafíos, especialmente si la hoja está doblada, tiene deformaciones o se encuentra ligeramente inclinada en la imagen. Estas imperfecciones generan inexactitudes en la identificación del contorno, que resultan en límites imprecisos o incluyendo información externa a la hoja. El programa trata de resolver este problema reduciendo el valor *epsilon* de la función *ApproxPolyDP* para controlar la precisión de la aproximación al polígono. No obstante, estas correcciones pueden no ser suficientes en casos de deformación pronunciada. Por lo tanto, se recomienda al usuario tomar la foto de la hoja lo más plana posible, sobre una superficie igualmente plana para minimizar las distorsiones en la imagen.

4. Transformación de perspectiva

Utilizando los vértices identificados en el contorno de la hoja, se lleva a cabo una transformación de perspectiva. Este proceso implica asociar los vértices de la hoja con los vértices de una hoja tamaño carta estándar mediante una transformación lineal de la imagen original. Para esto es necesario conocer el lado más largo de la hoja identificada y asociarlo con el lado más largo de una hoja tamaño carta. De esta forma, es posible eliminar todo el contenido exterior a la hoja, dando por resultando una imagen recortada y enfocada únicamente en el eslabón binario y la hoja de referencia.

Para la realización de este proceso, se obtienen las coordenadas de los vértices de la hoja carta con las dimensiones reales de dicha hoja. Utilizando estos puntos de referencia, se aplica la función *getPerspectiveTransform*, que calcula la matriz de transformación necesaria para la corrección de perspectiva. A continuación, se utiliza la función *warpPerspective*, capaz de aplicar la transformación de perspectiva a la imagen original mediante la matriz previamente calculada.



Ilustración 16.1. Fig 1



Ilustración 16.2. Fig 2



Ilustración 16.3. Fig 3

5. Arreglos de imagen y binarización

Si todos los pasos anteriores se han ejecutado correctamente, la imagen resultante deberá presentar únicamente la hoja alineada, lo cual permite avanzar hacia la identificación de características del eslabón.

Para este propósito, se realiza una conversión de la imagen alineada a escala de grises, seguida de un filtro de desenfoque gaussiano con un kernel 5x5 para reducir el ruido presente y suavizar la imagen. Luego, se emplea una umbralización Otsu para la binarización de la imagen, convirtiéndola en un complejo blanco y negro y permitiendo distinguir claramente el eslabón de la hoja. Después se realiza un arreglo morfológico



Ilustración 17.1. Fig 1

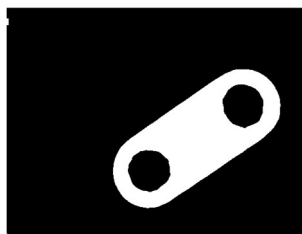


Ilustración 17.2. Fig 2



Ilustración 17.3. Fig 3

conocido como *Opening* con un kernel de 3x3 y 2 iteraciones. De esta forma se eliminan los posibles puntos aislados o detalles no deseados generados durante la binarización. El resultado final puede ser dos tipos: fondo negro con el eslabón en blanco, o fondo blanco con el eslabón negro, dependiendo de los colores predominantes en la imagen original.

6. Obtención de los contornos del eslabón

El objetivo de este punto es determinar las medidas principales del eslabón binario: la distancia entre centros de los agujeros de los extremos, el diámetro de los agujeros y el ancho del eslabón (ver ilustración 3). Este procedimiento se basa en el uso de momentos de imagen como se detalla en la sección sobre momentos en el marco teórico. Para realizar este análisis se emplea la función *moments* de OpenCV, capaz de aplicar la ecuación 1.A sobre la imagen ya binarizada en el paso anterior. Esta función opera con arreglos como los proporcionados por los contornos de una imagen, por ello se usa *findContours* previamente para hallar los contornos propios de los círculos de los nodos y el contorno exterior del eslabón.

Si la binarización previa no se ha realizado correctamente, pueden aparecer contornos adicionales debido a características internas o defectos del eslabón. Para mitigar este riesgo se ha optado por considerar únicamente los tres contornos con mayor cantidad de vértices identificados mediante *approxPolyDP*. Esta decisión se basa en la suposición de que los contornos asociados a los agujeros serán círculos, detectados con una cantidad significativa de vértices, mientras que otras imperfecciones generarían contornos más simples con pocos vértices.



Ilustración 18.1. Fig 1

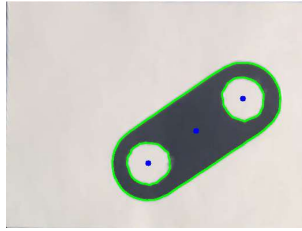


Ilustración 18.2. Fig 2

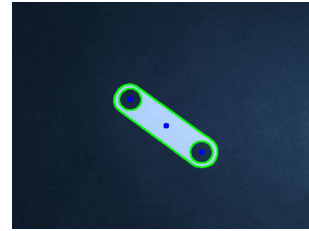


Ilustración 18.3. Fig 3

7. Encontrar las medidas características del eslabón

Una vez identificados los 3 contornos, se emplea *moments* para calcular los momentos geométricos de orden 0 y 1. Estos momentos permiten hallar el centroide de los contornos y su área con base en la ecuación 1.1. Utilizando los centroides de los agujeros de los nodos del eslabón, se determina directamente la distancia entre los centros para su caracterización. El área de cada contorno será empleada en una red neuronal (RN1) para estimar el radio interno de los nodos y el ancho del eslabón.

Red neuronal RN1

La red neuronal RN1 se ha creado mediante TensorFlor y Keras, con base en un algoritmo que define parámetros de entrada y salida, usados como arreglos para su entrenamiento. Los parámetros de salida son números aleatorios para el diámetro interior y ancho del eslabón. Mientras que los parámetros de entrada son la distancia entre centros y las áreas por contorno. El set de entrenamiento fue hecho con base en las ecuaciones 3 y 4.

Este modelo consta 3 capas densas con 120, 60 y 30 unidades respectivamente, cada una con una función de activación ReLU. Se utilizó la función de pérdida *mean_squared_error* y el optimizador *Adam* durante el entrenamiento. RN1 fue entrenada con 10000 datos de entrenamiento durante 500 epochs, utilizando un tamaño de lote 32 y reservando el 20% para validación. El resultado de esta estructura

fue resultado de un consecutivo cambio de capas con miras en minimizar al máximo la función de perdidas.

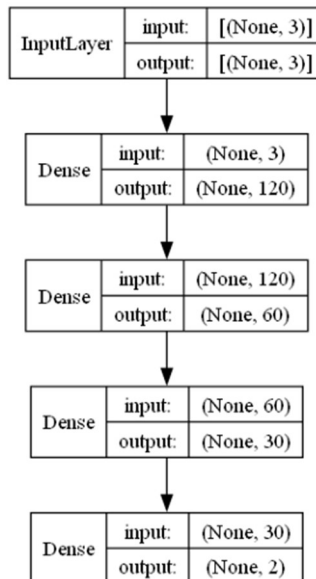


Ilustración 19. Estructura de la red neuronal RN1

Al finalizar el entrenamiento, se obtuvo una pérdida para los datos de validación de 0.0038 y una precisión de 100%.

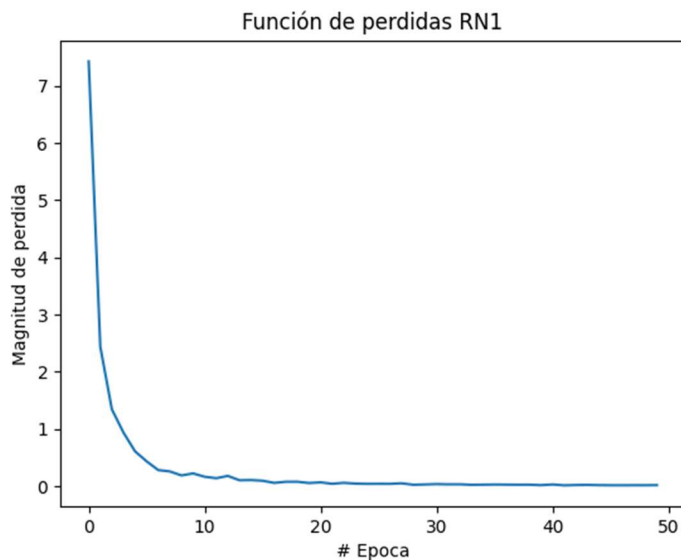


Ilustración 20. Función de pérdida para la red neuronal RN1 para los primeros 50 epochs

Al finalizar el entrenamiento, se obtuvo una función de pérdida (promedio de la diferencia al cuadrado entre las predicciones y las observaciones reales) (Big Data, 2019), para los datos de validación de 0.0555 y una precisión del 100%.

8. Estimar modos de falla

Una vez identificadas las dimensiones clave del eslabón binario, se puede avanzar en la determinación de los posibles modos de falla asociados a la figura analizada. Esto será posible después de que el programa conozca datos adicionales sobre características del eslabón, con los cuales poder estimar las fuerzas críticas y factores de seguridad de este para cada modo de falla, utilizando como base la red neuronal **RN2**.

Datos extra requeridos

Como se estableció en el marco teórico, para encontrar los posibles modos de falla, el programa requiere información adicional proporcionada por el usuario. Se debe ingresar:

- Espesor del eslabón
- Diámetro del pasador: Debe ser menor o igual al diámetro interno.
- Fuerza a la que estará sometido: Se considerará una fuerza aplicada en sentido de tensión, totalmente perpendicular al área transversal del eslabón, con el objetivo de restringir el análisis.
- Material del eslabón: Se supone idéntico al del pasador. Se ofrecen tres opciones de material para seleccionar:

Propiedades	Acero inoxidable 304	Aluminio 6061	Latón
S_{ut} (MPa): Resistencia a la tracción	515	310	425

S_{us} (MPa): Resistencia al corte	590	276	330
S_{bru} (MPa): Resistencia al rodamiento	300	70	140
E (GPa): Módulo de elasticidad	193	68	96

Tabla 2. Materiales posibles para el eslabón

Para ingresar esta información, el usuario dispone de la segunda ventana de la interfaz denominada: “Resultados del Análisis Dimensional”, que además contiene la forma y dimensiones del eslabón binario ingresado.



Resultados del Análisis Dimensional

Análisis de eslabones

Presione aquí si desea conocer el paso a paso de estos cálculos.

El análisis de eslabones se hizo mediante arreglos morfológicos a la imagen original. Presione aquí para continuar con el análisis de fallas.

Para más detalles consulte la guía.

Medidas útiles del eslabón:
Dc: 9.444 cm
Do: 2.082 cm
Ancho: 4.034 cm

5 coordenadas

Espesor: Ingrese el valor del espesor del eslabón binario en (cm):

Dímetro del pin: Ingrese el valor del diámetro del pin en (cm):

Fuerza: Ingrese el valor de la fuerza del eslabón binario en (kN):

Seleccione el material del eslabón:

Acero inoxidable 304

Ilustración 21. Ventana intermedia de la Interfaz para ingresar datos extra requeridos.

Red neuronal RN2

La red neuronal RN2 fue entrenada con valores aleatorios representativos para el espesor, diámetro del pasador, diámetro interno y ancho. Además, se incorporaron las propiedades del material del eslabón, elegido entre las opciones proporcionadas (véase

Tabla 2). Luego, utilizando las ecuaciones 5.1, 6.1, 7.1 y 8.1, se calculan las fuerzas críticas asociadas a cada posible modo de falla, que corresponden a los datos de salida.

Para estimar estos cálculos, **RN2** está diseñada con una estructura de capas densas. Comienza con una capa que contiene siete unidades correspondientes al espesor del eslabón (t), diámetro del pasador (D_p), diámetro interno (D_0), ancho (W) y las propiedades del material seleccionado S_{us} , S_{bru} y S_{ut} .

Después de una serie de cambios en busca de una red neuronal optima, que implicó diferentes estructuras y relaciones interneuronales, se optó por una red de 5 capas ocultas, que disminuyen progresivamente desde 200 hasta 4 unidades. Cada capa utiliza la función de activación ReLU, a excepción de la capa de salida que emplea una activación lineal. También se ha utilizado la función de pérdida *mean_squared_error* junto con el optimizador *Adam* para el entrenamiento de la red neuronal. Entrenamiento que se realizó con un set de 10000 unidades, 1000 epochs, un tamaño de lote 32 y reservando el 20% de los datos de entrenamiento para validación.

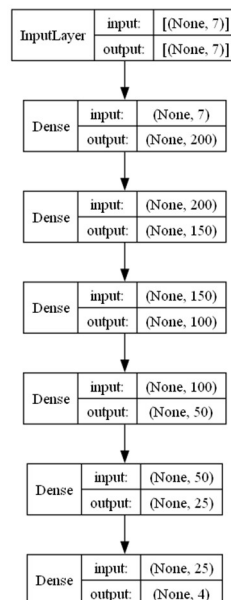


Ilustración 22. Estructura de la red neuronal RN2

Una vez entrenada RN2, para los datos de validación se obtuvo una magnitud de pérdidas de 97 y una precisión de 100%

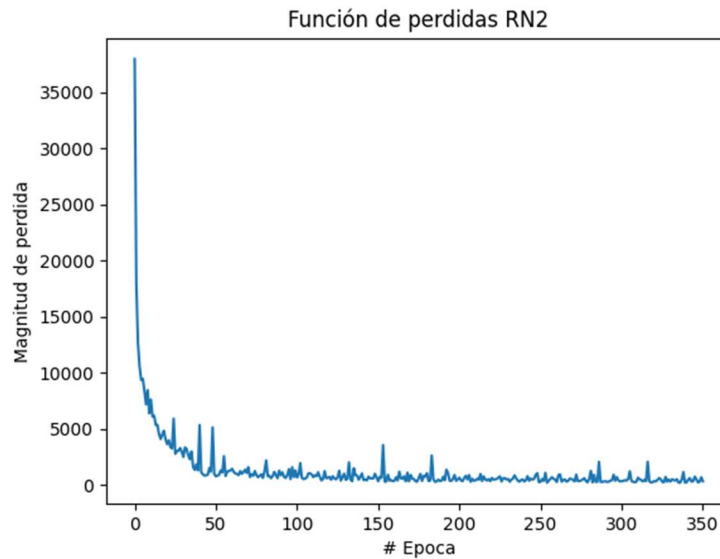


Ilustración 23. Función de pérdidas para la red neuronal RN2

Los datos proporcionados por la red neuronal luego son usados para el cálculo del factor de seguridad por cada modo de falla, empleando las ecuaciones 5.2, 6.2, 7.2 y 8.2.

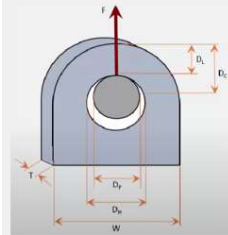
Visualizar modos de falla

Los resultados son mostrados a través de la tercera y última ventana del programa denominada: "Resultados del Análisis de Fallas"

Analisis de fallas y fuerzas criticas

El analisis de fuerzas se hizo mediante una red neuronal desarrollada de 1 dimension.
Para más detalles consulte la guía.

El eslabón seleccionado por usted tiene las siguientes características:

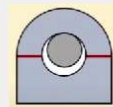


Características dimensionales del eslabón:

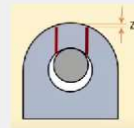
- Fapp: 50.0 kN
 - W: 4.034 cm
 - t: 1.0 cm
 - D1: 2.332 cm
 - D2: 2.0 cm
 - D3: 1.562 cm
 - D4: 4.034 cm
 - D5: 4.034 cm
 - D6: 4.034 cm
 - Z (Pérdida de longitud en el plano de corte): 0.1 cm
- Propiedades del material: Acero inoxidable 304
Supondremos que el pin es del mismo material que el eslabón.
- Resistencia a la tracción (Sut): 515.0 MPa
 - Resistencia al corte (Ssu): 590.0 MPa
 - Resistencia al rodamiento del eslabón (Sbru): 300.0 MPa
 - Modulo de elasticidad (E): 193.0 GPa

Para cada modo de falla se encontraron los siguientes resultados:

Falla de tensión en la sección neta
Fuerza crítica (Fc): 78.3597946105922 kN
Factor de seguridad (FS): 1.507



Falla por doble corte
Fuerza crítica (Fc): 330.096870953125 kN
Factor de seguridad (FS): 6.602



Falla de cizallamiento
Fuerza crítica (Fc): 62.61145401009786 kN
Factor de seguridad (FS): 1.252



Falla por hendidura
Fuerza crítica (Fc): 137.1890106201172 kN
Factor de seguridad (FS): 2.744

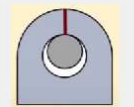


Ilustración 24. Última ventana de la interfaz, para ver los modos de falla.

En esta ventana primero se muestra el total de las dimensiones del eslabón basado en la ilustración 6 y las propiedades del material elegido. Luego se muestra por cada modo de falla la estimación de la fuerza crítica según la red neuronal **RN2**, para finalmente mostrar el factor de seguridad del eslabón con base en la fuerza dada por el usuario.

RESULTADOS

Para las figuras 1, 2 y 3, el programa dedujo sus dimensiones características de las que luego creó el siguiente bosquejo por eslabón:



Ilustración 24.1. Fig 1



Ilustración 24.2. Fig 2

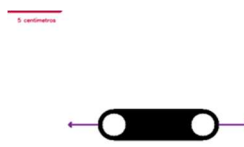


Ilustración 24.3. Fig 3

Dimensiones características

Las dimensiones características para cada eslabón son medidas mediante tres formas: utilizando el programa y algoritmo mostrado a lo largo del informe, una regla estándar cuya precisión es de $\pm 0.1\text{cm}$ y un calibrador pie de rey con una precisión de $\pm 0.02\text{cm}$:

Eslabones analizados	Dimensiones características del eslabón medidas con el Analizador de figuras								
	Distancia entre centros (cm)			Diámetro interior (cm)			Ancho (cm)		
	Programa	Regla	Calibrador	Programa	Regla	Calibrador	Programa	Regla	Calibrador
Figura 1	9,44366	9,5	9,46	2,08214	2,1	2,00	4,03441	4,0	4,00
Figura 2	10,6298	10,5	10,50	4,0063	3,9	4,00	6,97073	7,0	6,98
Figura 3	8,19183	8,2	8,14	2,18949	2,2	2,14	2,97772	3,0	3,08

Tabla 3. Comparación de resultados en la caracterización dimensional de cada eslabón analizado

Esta tabla evidenció una dificultad adicional al medir la distancia entre centros y el diámetro interno. Medir la distancia entre centros implica asumir los centroides de los pasadores, tanto para regla como para el calibrador. Además, la medición del diámetro interior se ve especialmente desafiante al utilizar la regla, ya que no ofrece la misma precisión y seguridad que el calibrador, el cual dispone de métodos para medir superficies interiores, como en este caso.

La tabla 3 hace parte de un conjunto más grande de datos presente en la sección de Anexos (Tabla 6), que incluye un ejercicio similar realizado para 40 imágenes adicionales, abarcando 9 eslabones binarios distintos. El propósito de esta tabla extendida es identificar el comportamiento general del programa al analizar una amplia gama de imágenes. Esto permitirá evaluar si el programa ofrece una mejor aproximación de las dimensiones de una pieza después de múltiples mediciones a la misma desde diferentes imágenes con ángulos y posiciones sobre la hoja.

Los resultados de tabla 5 muestran una diferencia promedio asombrosamente baja entre las mediciones efectuadas por el programa y las realizadas con un calibrador estándar:

- Diferencia media para la distancia entre centros: 0.068 *cm*
- Diferencia media para el diámetro interior: 0.0074 *cm*
- Diferencia media para el ancho del eslabón: 0.0041 *cm*

Esto sugiere una precisión notablemente alta en las mediciones obtenidas a través del programa, lo cual sugiere un rendimiento muy prometedor.

De igual forma, se calculó la desviación estándar de cada conjunto de imágenes para un mismo eslabón binario. Los resultados revelan una tendencia apreciable en la variabilidad de las mediciones:

- $\sigma_{Dist_C} = 0.216$
- $\sigma_{D_0} = 0.0770$
- $\sigma_W = 0.060$

Estos valores sugieren que la precisión previamente mostrada requiere de una muestra significativa de imágenes para el análisis. Es decir, si bien una sola imagen puede que no refleje con precisión las dimensiones características comparadas con un calibrador, la utilización de varias de ellas si lo hará, dado que el promedio de múltiples mediciones tiende a converger hacia las dimensiones reales.

Es de destacar que la desviación estándar es más alta para la distancia entre centros en comparación con las otras dimensiones. Esto podría deberse tanto a dificultades al medirla con el calibrador o a alguna complejidad inherente al análisis del programa. Esto significa que la distancia entre centros requiere de un mayor número de imágenes para obtener mediciones cercanas a la realidad, mientras que el ancho parece ser la dimensión que requiere de menos imágenes.

Modos de falla

Para presentar los modos de falla de cada eslabón propio de las figuras 1, 2 y 3 se hizo que cada uno tuviera un espesor de 1 cm y un diámetro de pin equivalente al diámetro interior del mismo. Todos estos eslabones están sometidos a una fuerza de 50 kN y están hechos de acero inoxidable 304, aluminio 6061 y latón respectivamente. La tabla 4 presenta las fuerzas críticas (F_c) estimadas y los correspondientes factores de seguridad (FS) para cada modo de falla de los eslabones analizados en las figuras mencionadas.

	Modo de falla 1		Modo de falla 2		Modo de falla 3		Modo de falla 4	
	F_c (kN)	FS	F_c (kN)	FS	F_c (kN)	FS	F_c (kN)	FS
Figura 1	78,36	1,5672	330,1	6,602	62,61	1,2522	137,18	2,7436
Figura 2	102,67	2,0534	325,46	6,5092	70,12	1,4024	179,64	3,5928
Figura 3	27,47	0,5494	138,38	2,7676	17,78	0,3556	74,94	1,4988

Tabla 4. Comparación de resultados en los modos de falla

Los resultados evidencian que la resistencia a la falla está relacionada con el tamaño de los eslabones. Es natural pensar que eslabones de menor tamaño serán capaces de resistir menores fuerzas y, por tanto, la pieza tendrá un menor factor de seguridad por modo de falla. No obstante, la figura 2, aunque de mayor tamaño, muestra ciertas particularidades. Por ejemplo, en el modo de falla 2, no presenta una fuerza crítica superior a la de los demás eslabones, debido a la resistencia del aluminio, muy inferior a la del acero inoxidable y el latón (véase tabla 2).

La fiabilidad de los valores de fuerza crítica se asocia con la precisión de la red neuronal **RN2**. La desviación cuadrada promedio entre predicciones y observaciones reales, mencionado en la sección de metodología “*Red neuronal RN2*”, fue de aproximadamente $97kN$ en 250 mediciones. Dicho otra forma, la red neuronal tiene una desviación de poco menos de $10kN$, lo que, aunque no es completamente exacto, refleja fielmente los modos de falla del eslabón, que es el objetivo central del proyecto.

Igual que las tablas 3 y 6, la tabla 4 es producto de una tabla más elaborada visible en la sección de anexos (tabla 7). Dicha tabla muestra los modos de falla para 43 imágenes analizadas bajo condiciones similares de tamaño, fuerza y materiales. De esta tabla se observan tendencias interesantes: el modo de falla 2 es el menos recurrente y requiere una fuerza crítica bastante elevada, sugiriendo su incidencia después de otro tipo de falla. Además, el modo de falla más común es el 3, asociado directamente con materiales para el eslabón como el Latón o Aluminio y piezas pequeñas dimensionalmente. Aproximadamente en el mismo nivel se encuentra el modo de falla 1, detectado únicamente con el aluminio y relacionado con un valor de distancia al borde D_l pequeño. Cabe aclarar que esto ocurre únicamente para los 9 eslabones analizados, por lo que no se descarta que las recurrencias aquí mostradas sean distintas para eslabones con dimensiones diferentes.

Si nos percatamos en las dimensiones que influyen en estos comportamientos, el analizador de figuras revela que un espesor mayor favorece una mayor fuerza crítica en cada modo de falla, y viceversa. Además, un diámetro de pasador similar al diámetro

interior parece no afectar la integridad del eslabón, a diferencia de un pasador más pequeño, que incide directamente en la ocurrencia del modo de falla número 3. Estas observaciones, respaldadas por el programa, pueden ser verificadas fácilmente a través de las ecuaciones 5, 6, 7 y 8.

No obstante, un diámetro de pasador superior al diámetro interior también es calculable por el programa, de hecho, elevarlo lo máximo posible también favorece una mayor fuerza crítica en el modo de falla 3. Aunque esto es una práctica en varias conexiones, no es práctico analizarlo en el programa dada la simplicidad del análisis realizado. Esta limitación se informa al usuario, ya que los resultados en tal caso serán erróneos

Costo computacional

Con el objetivo de evaluar la eficiencia del programa en cada etapa de análisis, se ha realizado un seguimiento del tiempo que el software emplea en distintas secciones. Para esto, se usó la librería Time de Python. Después de algunas iteraciones, se ha observado que, independientemente del eslabón bajo análisis, los tiempos por etapa son consistentes. A continuación, se presenta una tabla que resume el tiempo aproximado, en segundos, que el programa dedica a cada sección:

Sección	Tiempo promedio (segundos)
Importar bibliotecas	8 – 12
Procesamiento de la imagen	0.25 – 0.5
Funcionamiento de RN1	0.25
Funcionamiento de RN2	0.15
Mostrar ventanas de la interfaz	0.05 – 0.15

Tabla 5. Tiempos del software por sección

La carga inicial de las bibliotecas Tensorflow, OpenCV es el proceso más intensivo en términos de tiempo, sin embargo, se lleva a cabo una sola vez al inicio del programa. Esto

significa que, una vez cargadas, las bibliotecas permanecen accesibles durante toda la ejecución del análisis, sin importar si el usuario cambia de imagen, hasta que salga de la aplicación. Las etapas siguientes, como el procesamiento de la imagen y el funcionamiento de las redes neuronales demandan un tiempo considerablemente menor. Esto implica un funcionamiento fluido y rápido del programa durante su ejecución, que dependen del sistema y las especificaciones del hardware utilizadas.

DISCUSIÓN

La robustez del programa se evaluó mediante un conjunto de 100 imágenes, en donde el programa no mostró ningún error en 88% de las imágenes procesadas. Además, logró estimar con un margen de error inferior a 0.5cm respecto a las mediciones realizadas con un calibrador, las dimensiones características en el 85% de esas imágenes. Esto demuestra una precisión notable, especialmente cuando se utilizan múltiples imágenes del mismo eslabón. De hecho, con tan solo 10 imágenes del mismo eslabón, los resultados medidos para la distancia entre centros se alejan de los de un calibrador con precisión de 0.02cm en poco más de medio milímetro, mientras que, para el ancho y diámetro interior, es inferior a una centésima de milímetro. Asimismo, logra predecir con un margen de error de 10kN la fuerza con la que el eslabón fallará para cada uno de sus modos de falla.

Los contados errores del programa ocurren en su mayoría por defectos durante el procesamiento de imágenes, relacionados en gran parte en como se toma la foto para el análisis y algunos pasos de la metodología. Esto lleva a la necesidad de que el usuario sea consciente de los siguientes puntos al utilizar el problema:

- Contorneo de la hoja:

Una captura lejana o una falta de enfoque complican la identificación de los bordes de la hoja de referencia, lo cual resulta en errores durante el proceso de contorneo de esta. Como consecuencia, en varias ocasiones hay una reducción o

distorsión de las dimensiones del eslabón analizado, afectando la fiabilidad de las estimaciones realizadas por el programa.

- Identificar contornos:

El programa es especialmente sensible ante condiciones específicas en las imágenes, como brillos excesivos, sombras pronunciadas o zonas con falta de contraste. Esto puede dificultar la correcta identificación de los contornos de la hoja y el eslabón, que a veces se traduce en la imposibilidad de leer la imagen o generando errores durante el procesamiento.

Mismo hecho ocurre si hay una hoja adicional u otro elemento cuadriculado en la imagen o inclusive si hay manchas en la hoja, pues pueden ser identificados como contornos previos a visualizar el objeto que se pretende analizar.

- Limitaciones del Hardware:

El programa requiere de una calidad mínima de la cámara utilizada para obtener resultados precisos. Dispositivos con cámaras borrosas o incapaces de enfocar no podrán capturar imágenes que satisfagan los criterios mínimos de contraste y color necesarios para un análisis preciso.

De los 12 errores hallados en las imágenes de pruebas, 10 se relacionaban al paso 3 de la metodología, pues implicaban algún sector de la imagen que, por culpa del brillo o la sombra, no era identificado como parte de la hoja de referencia. Los otros 2 errores se produjeron en el paso 5, por culpa de un brillo excesivo sobre el contorno del eslabón. También, de las 13 imágenes donde hubo una medida muy alejada para las dimensiones del eslabón, 9 fueron por un contorno equivocado de la hoja, 3 por que los contornos del eslabón no fueron bien marcados y 1 porque se identificó un contorno en la hoja que no correspondía al eslabón.

Cabe resaltar ciertos aspectos problemáticos del proyecto, como su desarrollo limitado a una única forma de eslabón que solo permite cambiar la magnitud de sus

medidas. Quiere decir que ni siquiera es posible agregarle distintivos como dibujos, cortes o colores extra a un eslabón. Pues esto implican contornos extra analizados por el programa con los que no he sabido lidiar. De igual modo, también toca repasar la identificación de la hoja, al ser el paso que más presenta fallas, implica unos arreglos optimizables que para un proyecto escalable deben ser superadas.

Ampliar el proyecto hacia una mayor diversidad y cantidad de imágenes y objetos debe ser el siguiente paso para su evolución. Esto podría considerar la incorporación de técnicas de aprendizaje automático más sofisticadas que le permitan al programa generalizar patrones a partir de un conjunto amplio y variado de imágenes ya sea mediante redes neuronales convolucionales (CNN) por ejemplo o métodos de detección de características como SIFT (Scale – Invariant Feature Transform) o SURF (Speeded-Up Robust Features). Las cuales son capaces de identificar contornos y demás detalles de una imagen incluso bajo condiciones desafiantes como variaciones de iluminación, fondos complejos o distorsiones en la imagen. Esto implica que el mismo programa pueda identificar las dimensiones características del objeto a analizar y medirlas con el uso de la misma imagen, reduciendo la dependencia de técnicas como los momentos de imagen y probablemente mejorando la precisión y fiabilidad en la medición.

Otra opción podría ser la utilización de modelos pre-entrenados, diseñados para reconocer y localizar objetos de diferentes clases en imágenes como los que ofrece YOLO (You Only Look Once) o Faster R-CNN (Faster Region – based Convolutional Neural Networks). De esta forma sería posible reconocer primero dichos objetos usando estos modelos y luego aplicar el algoritmo desarrollado en este proyecto con los que ser capaces de caracterizarlo.

CONCLUSIÓN

El programa *analizador de figuras* desarrollado en Python, ha logrado cumplir con los objetivos propuestos al inicio de este informe. Es capaz de detallar el eslabón binario analizado al identificar sus dimensiones características y principales modos de fallas, junto con sus respectivas fuerzas críticas y factores de seguridad. Todo esto se lleva a cabo a través de una simple imagen del eslabón sobre una hoja tamaño carta, la cual es procesada por el programa mediante una interfaz interactiva fácil de utilizar.

La robustez del programa, probada mediante un conjunto diverso de imágenes, muestra una fiabilidad en la estimación de dimensiones que, aunque no supere por completo la precisión de instrumentos tradicionales, marca un camino prometedor. Además, su aplicación en la identificación de modos de falla y estimación de fuerzas críticas desarrollado exitosamente, subraya su potencial para predecir el rendimiento mecánico de los componentes con precisión, lo que puede tener severas implicaciones en el diseño y la ingeniería de sistemas mecánicos.

Si bien se han identificado desafíos y limitaciones, como la sensibilidad a ciertas condiciones de iluminación y la necesidad de mejoras en la captura de imágenes para garantizar resultados precisos y consistentes, estos hallazgos proporcionan una base sólida para futuras iteraciones y desarrollos. Esto sugiere posibles expansiones del programa hacia una gama más amplia de figuras, incluyendo aquellas de mayor complejidad dimensional.

El desarrollo de este proyecto representa un paso significativo en la exploración de alternativas automatizadas a la metrología convencional. Un área que seguramente estará en constante evolución, y las metodologías utilizadas aquí podrán servir como base para futuras investigaciones y aplicaciones similares en el campo.

BIBLIOGRAFÍA

- Big Data. (24 de 12 de 2019). *Funciones de pérdida en Machine Learning*. Obtenido de sitiobigdata.com: <https://sitiobigdata.com/2019/12/24/funciones-comunes-de-perdida-en-el-aprendizaje-automatico/>
- Engineering Solved. (17 de Julio de 2016). *4. Lifting Lug Analysis - Simplified*. Obtenido de youtube.com/@engineeringsolved2698: https://www.youtube.com/watch?v=oYzKEvcra8s&ab_channel=EngineeringSolved
- Garcia Garcia , P. P. (2013). *Reconocimiento de imágenes utilizando redes neuronales artificiales*. Madrid, España: DOCTA Complutense. Obtenido de Universidad Computense Madrid.
- Geeks For Geeks. (4 de Junio de 2023). *Find and Draw Contours using OpenCV | Python*. Obtenido de geeksforgeeks.org: <https://www.geeksforgeeks.org/find-and-draw-contours-using-opencv-python/>
- HWANG , W. M., & CHEN, H. M. (1995). *Synthesis of Kinematic Chains with Two Inversions*. Tainan, Taiwan: Department of Mechanical Engineering, National Cheng Kung University.
- IBM. (2012). *¿Qué es la inteligencia artificial?* Obtenido de ibm.com: <https://www.ibm.com/mx-es/topics/artificial-intelligence>
- IBM. (2014). *¿Qué es una red neuronal?* Obtenido de ibm.com: <https://www.ibm.com/topics/neural-networks>
- International Journal of Metrology and Quality Engineering (IJMQE). (5 de Octubre de 2019). *A brief history of metrology: past, present, and future*. Obtenido de metrology-journal.org: https://www.metrology-journal.org/articles/ijmqe/full_html/2019/01/ijmqe180023/ijmqe180023.html
- Luca, G. D. (11 de Noviembre de 2022). *Architecture: Criteria for Choosing the Number and Size of Hidden Layers*. Obtenido de baeldung.com: <https://www.baeldung.com/cs/neural-networks-hidden-layers-criteria>
- McCarthy, J. (12 de Noviembre de 2007). *WHAT IS ARTIFICIAL INTELLIGENCE?* Obtenido de Computer Science Department, Stanford University: <https://www-formal.stanford.edu/jmc/whatisai.pdf>
- MechaniCalc. (2014). *Análisis de terminales*. Obtenido de mechanicalc.com: <https://mechanicalc.com/reference/lug-analysis>
- Nowakowski, G., Dorogyy, Y., & Doroga-Ivaniuk, O. (2017). *Neural Network Structure Optimization Algorithm*. https://www.researchgate.net/publication/325755499_Neural_Network_Structure_Optimization_Algorithm: Journal of Automation, Mobile Robotics and Intelligent Systems.

OpenCV. (Junio de 2000). *About*. Obtenido de opencv.org: <https://opencv.org/about/>

Paradigma. (13 de Noviembre de 2017). *¿Es Python el lenguaje del futuro?* Obtenido de paradigmadigital.com: <https://www.paradigmadigital.com/dev/es-python-el-lenguaje-del-futuro/#:~:text=Podemos%20afirmar%20que%20Python%20es,lenguaje%20R%20es%20el%20rey.>

Sánchez, R. S. (Septiembre de 2008). *1º Tema. - Conceptos y definiciones en cinemática*. Obtenido de Universidad de Huelva, Escuela politecnica superior, Departamento de Ingeniería Minera, Mecánica y Energética:
<http://www.uhu.es/rafael.sanchez/ingenieriamaquinas/carpetaapuntes.htm/Apuntes%20Tema%201.pdf>

SYBRIDGE Technologies. (19 de Junio de 2020). *The Importance of Metrology in Manufacturing*. Obtenido de fastradius.com: <https://sybridge.com/metrology/>

TensorFlow. (2015). *Introducción a TensorFlow*. Obtenido de tensorflow.org:
<https://www.tensorflow.org/learn?hl=es-419>

Universidad de Sevilla. (2015). *Trabajo de Momentos*. Obtenido de grupo.us.es:
<https://grupo.us.es/gtocom/pid/pid10/doc.htm>

ANEXOS

Eslabones analizados	Distancia entre centros (cm)			Diámetro interior (cm)			Ancho (cm)		
	Programa	Regla	Calibrador	Programa	Regla	Calibrador	Programa	Regla	Calibrador
Figura 1	9,44366	9,5	9,46	2,08214	2,1	2,00	4,03441	4,0	4,00
Figura 1	9,29	9,5	9,46	1,99437	2,1	2,00	3,96176	4,0	4,00
Figura 1	9,4044	9,5	9,46	1,99773	2,1	2,00	3,9725	4,0	4,00
Figura 1	9,43995	9,5	9,46	1,9986	2,1	2,00	3,97408	4,0	4,00
Figura 1	9,75577	9,5	9,46	2,09549	2,1	2,00	4,05385	4,0	4,00
Figura 1	9,47844	9,5	9,46	2,09884	2,1	2,00	4,05453	4,0	4,00
Figura 2	10,6298	10,5	10,50	4,0063	3,9	4,00	6,97073	7,0	6,98
Figura 2	10,25246	10,5	10,50	3,72326	3,9	4,00	6,95556	7,0	6,98
Figura 2	10,84792	10,5	10,50	4,116	3,9	4,00	6,96885	7,0	6,98
Figura 3	8,19183	8,1	8,14	2,18949	2,2	2,14	2,97772	3,0	3,08
Figura 3	8,0046	8,1	8,14	2,13349	2,2	2,14	3,16457	3,0	3,08
Figura 3	8,10783	8,1	8,14	2,16108	2,2	2,14	3,05478	3,0	3,08
Figura 3	8,30977	8,1	8,14	2,15912	2,2	2,14	3,10312	3,0	3,08
Figura 3	8,16294	8,1	8,14	2,12604	2,2	2,14	3,16270	3,0	3,08
Figura 4	16,09032	16,0	16,06	4,71527	4,7	4,72	6,7058	6,7	6,76
Figura 4	15,92932	16,0	16,06	4,76115	4,7	4,72	6,78787	6,7	6,76
Figura 4	16,14687	16,0	16,06	4,8222	4,7	4,72	6,77429	6,7	6,76
Figura 5	20,12239	20,0	20,00	4,18955	4,0	4,00	6,07097	6,0	6,00
Figura 5	20,06554	20,0	20,00	4,12726	4,0	4,00	6,0545	6,0	6,00
Figura 6	11,40953	11,5	11,52	5,3726	5,5	5,52	7,83387	7,8	7,80
Figura 6	11,5385	11,5	11,52	5,3767	5,5	5,52	7,81082	7,8	7,80
Figura 6	11,80054	11,5	11,52	5,3845	5,5	5,52	7,78812	7,8	7,80
Figura 6	11,86856	11,5	11,52	5,49541	5,5	5,52	7,80427	7,8	7,80
Figura 7	12,67495	13,0	13,04	3,97685	4,0	4,08	5,35208	5,5	5,52
Figura 7	12,5113	13,0	13,04	3,87201	4,0	4,08	5,35296	5,5	5,52
Figura 7	13,60051	13,0	13,04	4,08502	4,0	4,08	5,55326	5,5	5,52
Figura 7	13,42207	13,0	13,04	4,16498	4,0	4,08	5,79056	5,5	5,52
Figura 7	13,12429	13,0	13,04	4,01092	4,0	4,08	5,5369	5,5	5,52
Figura 8	10,25273	10,5	10,48	4,32906	4,2	4,20	5,04179	5,0	5,00
Figura 8	10,42472	10,5	10,48	4,28603	4,2	4,20	5,03779	5,0	5,00
Figura 8	11,12937	10,5	10,48	4,18553	4,2	4,20	4,91938	5,0	5,00
Figura 8	10,92867	10,5	10,48	4,22925	4,2	4,20	4,91197	5,0	5,00
Figura 9	12,28362	12,0	12,14	5,80547	5,8	5,80	7,14006	7,2	7,22
Figura 9	12,28966	12,0	12,14	5,78263	5,8	5,80	7,34286	7,2	7,22
Figura 9	12,3031	12,0	12,14	5,82908	5,8	5,80	7,19274	7,2	7,22
Figura 9	12,01162	12,0	12,14	5,82078	5,8	5,80	7,18449	7,2	7,22
Figura 9	12,32947	12,0	12,14	5,91099	5,8	5,80	7,27549	7,2	7,22

Tabla 6. Mediciones de diferentes eslabones con diferentes imágenes

	Modo de falla 1		Modo de falla 2		Modo de falla 3		Modo de falla 4		Material
	Fc (kN)	FS	Fc (kN)	FS	Fc (kN)	FS	Fc (kN)	FS	
Figura 1	78,36	1,5672	330,1	6,602	62,61	1,2522	137,18	2,7436	Acero
Figura 1	60,15	1,203	174,01	3,4802	13,7	0,274	91,98	1,8396	Aluminio
Figura 1	82,15	1,643	196,67	3,9334	23,36	0,4672	121,49	2,4298	Latón
Figura 1	79,22	1,5844	328,95	6,579	62,04	1,2408	136,56	2,7312	Acero
Figura 1	78,68	1,5736	331,13	6,6226	62,68	1,2536	137,62	2,7524	Aluminio
Figura 1	83,2	1,664	201,55	4,031	24,23	0,4846	124,02	2,4804	Latón
Figura 2	102,67	2,0534	325,46	6,5092	70,12	1,4024	179,65	3,593	Aluminio
Figura 2	162,3	3,246	607,85	12,157	118,53	2,3706	256,52	5,1304	Acero
Figura 2	145,34	2,9068	373,66	7,4732	92,5	1,85	225,64	4,5128	Latón
Figura 3	27,47	0,5494	138,38	2,7676	17,78	0,3556	74,95	1,499	Latón
Figura 3	27,52	0,5504	119,3	2,386	14,59	0,2918	63,52	1,2704	Aluminio
Figura 3	33,604	0,67208	142,92	2,8584	18,5	0,37	79,24	1,5848	Latón
Figura 3	53,407	1,06814	239,24	4,7848	55,91	1,1182	92,24	1,8448	Acero
Figura 3	27,75	0,555	119,38	2,3876	14,61	0,2922	63,62	1,2724	Aluminio
Figura 4	114,91	2,2982	504,95	10,099	145,34	2,9068	225,39	4,5078	Acero
Figura 4	76,01	1,5202	332,34	6,6468	96,21	1,9242	177,5	3,55	Aluminio
Figura 4	115,04	2,3008	376,85	7,537	122,07	2,4414	217,19	4,3438	Latón
Figura 5	96,95	1,939	435,85	8,717	118,01	2,3602	193,06	3,8612	Acero
Figura 5	64,62	1,2924	282,11	5,6422	69,38	1,3876	144,88	2,8976	Aluminio
Figura 6	155,78	3,1156	643,87	12,8774	180,52	3,6104	285,83	5,7166	Acero
Figura 6	88,19	1,7638	398,83	7,9766	116,45	2,329	218,52	4,3704	Aluminio
Figura 6	131,31	2,6262	444,7	8,894	145,67	2,9134	264,17	5,2834	Latón
Figura 6	150,09	3,0018	632,26	12,6452	184,81	3,6962	282,2	5,644	Acero
Figura 7	83,15	1,663	355,89	7,1178	101,35	2,027	157,32	3,1464	Acero
Figura 7	51,75	1,035	240,3	4,806	56,63	1,1326	121,33	2,4266	Aluminio
Figura 7	89,57	1,7914	295,92	5,9184	85,8	1,716	165,29	3,3058	Latón
Figura 7	90,11	1,8022	395,45	7,909	109,25	2,185	175,35	3,507	Acero
Figura 7	56,02	1,1204	256,23	5,1246	63,72	1,2744	129,12	2,5824	Aluminio
Figura 8	80,13	1,6026	329,32	6,5864	100,12	2,0024	145,22	2,9044	Acero
Figura 8	49,9	0,998	234,52	4,6904	59,57	1,1914	113,59	2,2718	Aluminio
Figura 8	64,02	1,2804	261,94	5,2388	75,74	1,5148	134,78	2,6956	Latón
Figura 8	80,99	1,6198	330,71	6,6142	103,22	2,0644	146,28	2,9256	Acero
Figura 9	119,13	2,3826	487,17	9,7434	165,31	3,3062	221,89	4,4378	Acero
Figura 9	90,22	1,8044	397,13	7,9426	128,4	2,568	212,13	4,2426	Aluminio
Figura 9	115,55	2,311	428,61	8,5722	160,14	3,2028	244,43	4,8886	Latón
Figura 9	120,4	2,408	493,51	9,8702	166,82	3,3364	224,67	4,4934	Acero
Figura 9	75,23	1,5046	370,97	7,4194	99,12	1,9824	191,81	3,8362	Aluminio

Tabla 7. Modos de falla para diferentes eslabones