

## Sistema de calibración de impresoras 3D delta basado en visión artificial

### A Computer Vision-Based Delta 3D Printer Calibration System

Rubén Eduardo Aguilera-Pérez<sup>1</sup>, Irving Armando Cruz-Albarrán<sup>2</sup>, Jesús Rooney Rivera-Guillén<sup>1\*</sup>

<sup>1</sup> ENAP -Research Group, CA-Sistema Dinámicos y Control, Universidad Autónoma de Querétaro (UAQ), Campus San Juan del Río, Río Moctezuma 249, Col. San Cayetano, San Juan del Río, CP. 76807, México.

<sup>2</sup> Laboratorio de Visión Artificial y Termografía/Mecatrónica, Facultad de Ingeniería, Universidad Autónoma de Querétaro, Campus San Juan del Río, San Juan del Río, CP. 76807, México.

jesus.rooney.rivera@uaq.mx

\*Autor de correspondencia

#### Resumen

Las impresoras 3D de cinemática delta son conocidas por su velocidad, sin embargo, su calibración es un desafío complejo y dependiente del operario. Las soluciones automatizadas actuales, que usan sondas de contacto, no corrigen los errores dimensionales y angulares de raíz. En este trabajo se presenta un sistema de calibración automatizado basado en visión artificial diseñado para superar estas limitaciones. La metodología se centra en un patrón de calibración con marcadores ArUco que la propia impresora fabrica. Este patrón es analizado por el sistema de visión propuesto que, tras corregir la distorsión de la lente, mide con alta precisión la geometría real del objeto para calcular los ajustes del firmware. La validación experimental demuestra que el sistema propuesto reduce el tiempo de calibración en un 50% (de 2 a 1 minuto) en comparación con un método manual. Aunque el error relativo medio es de 0.75 % y el error absoluto promedio es de 0.7476 mm debido a su propia definición, el principal avance es la creación de un método objetivo, rápido y de bajo costo que reduce la subjetividad y la curva de aprendizaje del proceso manual, estableciendo un marco accesible para la calibración de alta precisión.

**Palabras clave:** Impresora 3D; calibración Delta; visión artificial; marcadores ArUco; procesamiento de imágenes; manufactura aditiva.

#### Introducción

La proliferación de la manufactura aditiva, comúnmente conocida como impresión 3D, representa un pilar fundamental de la cuarta revolución industrial, o Industria 4.0. Esta tecnología ha alterado fundamentalmente el panorama de la fabricación al permitir la creación rápida de prototipos, la producción bajo demanda de geometrías complejas y la descentralización de las capacidades de manufactura (Clavel, 1988). Entre los diversos sistemas cinemáticos empleados en las impresoras de modelado por deposición fundida (FDM, por sus siglas en inglés), la arquitectura de robot delta destaca por sus ventajas únicas.

Las impresoras delta son máquinas cinemáticas paralelas caracterizadas por tres brazos conectados a un efector final central, donde cada brazo es impulsado por un carro que se traslada verticalmente. Este diseño mantiene los voluminosos motores paso a paso estacionarios, reduciendo drásticamente la masa móvil y la inercia del sistema (Pierrot & Company, 1999). El resultado es una máquina capaz de alcanzar velocidades y aceleraciones excepcionalmente altas, lo que la hace particularmente adecuada para aplicaciones de impresión rápida (Bonev, 2001; D. Zhang *et al.*, 2022). Esto contrasta marcadamente con los sistemas cartesianos tradicionales, donde uno o más motores deben moverse junto con el marco del extrusor, aumentando la inercia y limitando las velocidades alcanzables (Pierrot & Company, 1999), ventajas adicionales están en que, usando un sistema de extrusión Bowden, que el motor impulsor está lejos de la boquilla y el filamento es conducido por un tubo de teflón, lo cual baja mucho la masa del efector reduciendo la inercia con la que se tiene que lidiar aún más.

A pesar de las ventajas de rendimiento, la estructura cinemática paralela de las impresoras delta introduce un desafío significativo: la calibración. La relación entre la posición lineal de los tres carros de accionamiento y las coordenadas cartesianas resultantes (X,Y,Z) del efector final se rige por un conjunto complejo y no lineal de ecuaciones trigonométricas (Moser *et al.*, 2021). A diferencia de las impresoras cartesianas donde los ejes son en gran medida independientes, en una máquina delta, cada movimiento del efector requiere el



movimiento coordinado de las tres torres. En consecuencia, imprecisiones físicas diminutas en la geometría de la impresora —como la longitud de los brazos diagonales, el espaciado preciso de las torres verticales o las posiciones de los topes finales (endstops)— tienen un efecto compuesto y no intuitivo en la geometría final de la impresión. Esta complejidad inherente hace que la calibración sea una tarea no trivial que impacta directa y profundamente en la precisión dimensional, la adhesión de la primera capa y la fiabilidad y calidad general de los objetos impresos (Klipper, 2023b; Marlin Firmware, 2023; RepRap, 2023a). Una impresora delta incorrectamente calibrada producirá piezas con dimensiones distorsionadas, superficies curvas que deberían ser planas y una mala adhesión a la cama, inutilizando la máquina para trabajos de precisión.

Esta complejidad inherente y la pronunciada curva de aprendizaje asociada a la calibración manual son factores clave que han limitado la proliferación de las impresoras delta en comparación con los sistemas cartesianos, a pesar de sus evidentes ventajas en velocidad. El proceso tradicional depende en gran medida de la habilidad y paciencia del operador, lo que presenta una barrera de entrada significativa para los usuarios novatos y sigue siendo una fuente de ineficiencia incluso para los más experimentados.

Con el tiempo, se han desarrollado varios métodos para abordar el desafío de la calibración, cada uno con un conjunto distinto de ventajas y limitaciones. La evolución de estas técnicas refleja una tendencia más amplia en la tecnología de fabricación, pasando de la artesanía dependiente del operador hacia la automatización determinista y, en última instancia, a sistemas inteligentes basados en datos.

El enfoque más básico para la calibración es un proceso manual e iterativo de "prueba y error". Esto típicamente implica que un operador imprima una serie de objetos de prueba, como las estrellas de calibración, los mida con calibradores (calipers) y luego ajuste manualmente los parámetros cinemáticos en el firmware de la impresora. El ajuste fino del desplazamiento de la altura Z y el nivel de la cama a menudo se basa en retroalimentación táctil subjetiva, como la "prueba del papel", donde se utiliza una hoja de papel para medir la fricción entre la boquilla y la cama de impresión en varios puntos (Ganitano *et al.*, 2025; Klipper, 2023a; Marlin Firmware, 2023), y se puede cambiar por el uso de un calibrador de láminas, lo cual da un poco más de precisión. Aunque es capaz de producir buenos resultados en manos de un operador experto, este método consume excepcionalmente mucho tiempo, carece de repetibilidad y depende en gran medida de la habilidad y paciencia del operador, mientras que un operador experimentado puede realizar una calibración casi perfecta, un novato sólo podrá hacer una apenas aceptable. Esta curva de aprendizaje pronunciada presenta una barrera de entrada significativa para los usuarios novatos y sigue siendo una fuente persistente de frustración e ineficiencia incluso para los profesionales más experimentados (RepRap, 2023b).

Para superar las inconsistencias de los métodos manuales, las rutinas de calibración automatizadas que utilizan sondas de contacto físicas se han vuelto comunes. Sistemas como el BLTouch, sensores inductivos o resistores sensibles a la fuerza (FSR, por sus siglas en inglés) integrados en la cama de impresión permiten que el firmware de la impresora mida automáticamente la altura de la cama en una serie de puntos predefinidos (Kang *et al.*, 2023; Moser *et al.*, 2021). El firmware luego utiliza estos datos para calcular y aplicar correcciones a los parámetros cinemáticos. Esto representa un avance significativo, reemplazando el juicio humano subjetivo con un proceso mecánico más determinista.

Sin embargo, estos sistemas basados en contacto no están exentos de sus propias limitaciones mecánicas fundamentales. La propia sonda puede convertirse en una fuente de error. Las sondas mecánicas pueden tener un sesgo de ubicación inherente que puede exceder los 25 micrones (0.025 mm), un error significativo en el contexto de la impresión de precisión (Moser *et al.*, 2021). Además, en una impresora delta, la fuerza requerida para activar la sonda puede hacer que el efector final se incline ligeramente, ya que no está rígidamente restringido de la misma manera que un efector cartesiano, ya que las deltas tienen cierto juego que les permite moverse más rápido en sus articulaciones, y que normalmente usan a la gravedad para estar en su posición correcta. Esta inclinación puede sesgar la medición, lo que lleva a un mapa de calibración inexacto (Moser *et al.*, 2021). Diferentes tecnologías de sonda tienen modos de falla únicos; por ejemplo, los FSR pueden atascarse o exhibir presiones de activación variables dependiendo de la temperatura y el estrés mecánico (Kang *et al.*, 2023). Crucialmente, estos sistemas miden un estado intermedio de la máquina —la distancia percibida a la cama— en lugar del resultado real del proceso de impresión. Por lo tanto, su precisión está fundamentalmente limitada por la precisión y fiabilidad de sus propios componentes mecánicos.



En este artículo se propone un sistema de calibración basado en visión artificial que ofrece una solución semiautomatizada, de bajo costo y simple para la calibración en impresoras 3D delta. La tesis central es que, al medir ópticamente las propiedades geométricas de una figura de calibración impresa, el sistema propuesto puede medir automáticamente el tamaño y ángulo de las barras de este modelo para que sea un proceso que cualquier persona, independientemente de su experiencia, pueda realizar las calibraciones. El desempeño de la propuesta es validado experimentalmente, comparando el sistema automatizado contra una calibración manual experta. Los resultados demuestran que el método basado en visión ofrece una metodología consistente e independiente de la habilidad del operario, reduciendo el tiempo de calibración en un 50 %. Si bien la precisión dimensional inicial fue menor que la lograda por el operario experto, este enfoque establece un precedente para la calibración objetiva. Este enfoque, al aprovechar hardware de costo reducido como cámaras web de alta definición, tiene el potencial de lograr resultados de precisión excepcionalmente alta sin la necesidad de equipos costosos y especializados (Kang *et al.*, 2023; RepRap, 2023b).

## Principios Fundamentales y Trabajos Relacionados

Una comprensión profunda del sistema de calibración basado en visión propuesto requiere un conocimiento fundamental de la mecánica subyacente de las impresoras delta y el contexto de la investigación de calibración existente. Esta sección primero profundiza en el modelo matemático que gobierna el movimiento del robot delta y luego proporciona una revisión crítica de la literatura relevante.

## El Modelo Cinemático de los Robots Delta Lineales

La cinemática es la rama de la mecánica que describe el movimiento de los objetos sin considerar las fuerzas que causan el movimiento (Taufik & Jain, 2016). Para una impresora 3D delta, la cinemática define la relación matemática entre las posiciones de los tres carros que se mueven verticalmente y la posición del efector final (el hotend) en el espacio 3D. Esta relación es bidireccional, lo que conduce a dos problemas computacionales distintos pero relacionados: la cinemática inversa y la cinemática directa (Bonev, 2001; D. Zhang *et al.*, 2022).

### Cinemática Inversa

La cinemática inversa es el proceso de determinar las posiciones requeridas de los actuadores (es decir, las alturas de los tres carros en sus torres verticales) para lograr una posición deseada del efector final ( $X, Y, Z$ ) (Bonev, 2001). Este es el cálculo fundamental que el firmware de la impresora debe realizar continuamente para traducir un comando de G-code, como G1 X10 Y20 Z5, en movimientos específicos para los tres motores paso a paso.

La solución geométrica al problema de la cinemática inversa es relativamente sencilla debido al diseño simétrico de la impresora. El proceso se puede conceptualizar de la siguiente manera (Bonev, 2001):

1. **Definir el sistema:** la geometría se define por las posiciones de las tres torres verticales, la longitud de las varillas diagonales y las dimensiones del efector final.
2. **Proyectar en los planos de las torres:** para cada uno de los tres enlaces torre-brazo-efector, el problema puede simplificarse considerándolo en el plano 2D que contiene el eje vertical de la torre y el efector final.
3. **Resolver para la altura del carro:** En este plano 2D, la coordenada objetivo ( $X, Y, Z$ ) del efector final define un punto. La junta universal en el efector final está a una distancia horizontal conocida de este punto. La varilla diagonal actúa como la hipotenusa de un triángulo rectángulo, con una longitud conocida. El carro debe posicionarse a una altura en la torre tal que la varilla diagonal pueda conectar la junta universal del carro con la junta universal del efector final. Usando el teorema de Pitágoras, se puede calcular la altura vertical requerida del carro. En la ecuación (1) podemos ver la representación del teorema de Pitágoras para la delta, donde  $Z_A$  es la altura del carro,  $D$  es la longitud de la barra diagonal,  $x_A, y_A$  son las coordenadas de anclaje de la torre en el plano XY, y  $x_e, y_e, z_e$  son las coordenadas deseadas del efector.



$$Z_A = \sqrt{D^2 - (x_A - x_C)^2 - (y_A - y_C)^2} + z_e \quad (1)$$

4. **Repetir para todas las torres:** este cálculo se realiza de forma independiente para cada una de las tres torres, produciendo las alturas requeridas ( $Z_A$ ,  $Z_B$ ,  $Z_C$ ) para los carros en las torres A, B y C, respectivamente. El firmware luego ordena a los motores que se muevan a estas posiciones.

#### Definición de los Parámetros Geométricos

La precisión de los cálculos de cinemática inversa y directa depende enteramente de la precisión de los parámetros geométricos utilizados en el modelo. Estos parámetros se definen en el firmware de la impresora, y son estos valores los que la calibración busca optimizar. La Tabla 1 proporciona un léxico definitivo para estos parámetros críticos tal como se implementan comúnmente en firmwares como Marlin, conectando las variables de software con sus contrapartes físicas (Marlin Firmware, 2023). Una comprensión clara de estos parámetros es esencial, ya que son los objetivos directos del proceso de optimización en el sistema basado en visión propuesto.

**Tabla 1: Definición de Parámetros Cinemáticos Delta** (Klipper, 2023b; Marlin Firmware, 2023)

| Parámetro (Marlin)      | Definición   | Rol Geométrico e Impacto en la Calibración  |
|-------------------------|--|---|
| DELTA_DIAGONAL_ROD      | La distancia de centro a centro entre las juntas universales en los extremos de un solo brazo diagonal.  | Este parámetro gobierna directamente la escala general de los movimientos en el plano XY. Un valor incorrecto resulta en impresiones dimensionalmente inexactas; por ejemplo, una pieza diseñada para tener 100 mm de ancho puede imprimirse con 98 mm. Esta es una de las fuentes más comunes de error dimensional.  |
| DELTA_RADIUS            | La distancia horizontal efectiva desde el eje central de la impresora hasta el punto de pivote de una varilla diagonal en un carro cuando el efector final está centrado en (0,0,Z). | Este es uno de los parámetros más críticos para la planitud de la impresión. Un DELTA_RADIUS incorrecto hace que el efector final se desplace en una trayectoria curva en lugar de un plano. Si el radio es demasiado pequeño, la trayectoria es convexa (un "domo"), haciendo que la boquilla esté demasiado alta en el centro. Si es demasiado grande, la trayectoria es cóncava (un "cuenco"), haciendo que la boquilla esté demasiado alta en los bordes. |
| DELTA_SMOOTH_ROD_OFFSET | La distancia horizontal desde el eje central de la impresora hasta el centro de una guía lineal vertical (o extrusión).  | Esta es una medida física primaria que se utiliza en el cálculo del DELTA_RADIUS por parte del firmware.  |
| DELTA_EFFECTOR_OFFSET   | La distancia horizontal desde el centro del efector final hasta el centro de la junta  | Esta es otra medida física utilizada para calcular el DELTA_RADIUS efectivo.  |



|                        |   |  |
|------------------------|---|--|
|                        | universal de un brazo diagonal.   |  |
| DELTA_CARRIAGE_OF_FSET | La distancia horizontal desde el centro de una guía lineal vertical hasta el centro de la junta universal de un brazo diagonal en el carro.   | Esta es la tercera medida física utilizada en el cálculo del DELTA_RADIUS.   |
| DELTA_TOWER_ANGLE_TRIM | Un conjunto de valores de corrección que ajustan la posición angular asumida de cada torre en relación con la separación ideal de 120 grados. | Estas correcciones compensan las imperfecciones en el ensamblaje del marco de la impresora. Si las torres no son perfectamente equidistantes, las impresiones saldrán sesgadas, haciendo que los cuadrados se impriman como trapezoides o rombos. Este parámetro corrige tales distorsiones geométricas. |

## Una Revisión de las Metodologías de Calibración

La literatura y las prácticas de la comunidad en torno a la calibración de impresoras delta se pueden clasificar ampliamente en enfoques manuales, basados en sondas de contacto y basados en visión.

### *Procedimientos Manuales y Semiautomatizados*

Firmwares como Marlin y Klipper incluyen rutinas integradas para ayudar con la calibración. El comando G33 de Marlin, por ejemplo, puede automatizar el proceso de sondeo, pero finalmente depende del usuario para interpretar los resultados y tomar decisiones (Marlin Firmware, 2023). El comando DELTA\_CALIBRATE de Klipper guía al usuario a través de una serie de siete puntos de sondeo manual (centro y un patrón hexagonal) utilizando la prueba del papel (Klipper, 2023a). Si bien estas rutinas proporcionan un flujo de trabajo estructurado, a menudo solo logran una calibración gruesa. Lograr una alta precisión dimensional con frecuencia requiere un bucle secundario y totalmente manual de imprimir un objeto de prueba, medirlo con calibradores y ajustar finamente parámetros como DELTA\_DIAGONAL\_ROD para corregir errores de escala (Klipper, 2023a; RepRap, 2023b). Esto resalta una limitación fundamental: estos métodos semiautomatizados son excelentes para nivelar la cama (calibrar la planitud de la altura Z) pero son incapaces de medir o corregir errores en las dimensiones XY.

Como se discutió, las sondas automatizadas mejoran la repetibilidad, pero no son una panacea. Un análisis crítico revela una desconexión entre lo que miden estas sondas y las causas raíz de todos los errores dimensionales. Una sonda mide la desviación de la altura Z en varias coordenadas XY (Klipper, 2023a; RepRap, 2023b). El algoritmo de calibración luego ajusta parámetros —principalmente DELTA\_RADIUS y los desplazamientos de los topes finales— para minimizar esta desviación Z, forzando efectivamente a la boquilla a viajar en un plano. Sin embargo, este proceso no proporciona información sobre la precisión de los movimientos laterales. Para medir un error de escala XY (por ejemplo, una línea de 100 mm que se imprime como 99 mm), uno debe medir las dimensiones laterales de un objeto impreso (Kang *et al.*, 2023; Klipper, 2023a). Una sonda Z es físicamente incapaz de realizar esta medición. De manera similar, para detectar y corregir errores de ángulo de torre que causan sesgo, uno debe evaluar la "cuadratura" o las dimensiones relativas de un objeto impreso, otra tarea más allá de la capacidad de una sonda Z (Klipper, 2023b). En consecuencia, incluso una impresora que ha sido "perfectamente" calibrada con una sonda de contacto todavía puede producir piezas dimensionalmente inexactas, lo que requiere un paso de calibración manual posterior. Esto revela una brecha crítica en las soluciones automatizadas existentes que un sistema basado en visión está en una posición única para llenar.



La aplicación de la visión artificial a la impresión 3D es un campo de investigación en crecimiento. Es importante distinguir entre dos áreas de aplicación principales: el monitoreo durante el proceso y la calibración previa al proceso.

Un cuerpo significativo de trabajo se centra en el monitoreo durante el proceso para la detección de fallas por medio de visión artificial. Estos sistemas usan cámaras para observar el proceso de impresión en tiempo real, empleando algoritmos para detectar fallas comunes como problemas de extrusión de filamento ("espagueti"), desplazamiento de capas, deformación de la pieza fuera de la cama u otros defectos (Silva *et al.*, 2024; Temistocles *et al.*, 2023). El objetivo de estos sistemas es alertar al usuario o pausar automáticamente la impresión para ahorrar tiempo y material, en lugar de mejorar la precisión geométrica intrínseca de la máquina.

Un cuerpo de investigación más pequeño pero muy relevante aborda la calibración previa al proceso utilizando visión por computadora. Estos enfoques se alinean más estrechamente con el trabajo presentado en este artículo. Algunos investigadores han propuesto métodos que implican imprimir reglas u objetos de calibración personalizados y luego usar una cámara y procesamiento de imágenes para medir sus dimensiones automáticamente (Valiente *et al.*, 2021). Otros enfoques avanzados han explorado el uso del aprendizaje profundo, como las redes neuronales convolucionales (CNN), para realizar la detección de características en objetos impresos sin la necesidad de marcadores geométricos simples (Wu *et al.*, 2020). Algunos sistemas también han utilizado configuraciones estereoscópicas de múltiples cámaras para reconstruir un modelo 3D de la pieza impresa para su análisis (Kang *et al.*, 2023). Este trabajo existente valida el concepto fundamental de usar la visión para medir artefactos impresos para la calibración. Sin embargo, todavía existe la oportunidad de desarrollar un sistema que sea de bajo costo (que requiera solo una cámara económica), altamente robusto (que use marcadores fiduciales para una detección de características inequívoca) y holístico (capaz de corregir todos los parámetros cinemáticos principales simultáneamente). Este artículo tiene como objetivo llenar ese vacío.

## El Marco de Calibración Propuesto Basado en Visión Artificial

El núcleo de esta investigación es un marco integral que aprovecha la visión artificial para semiautomatizar y mejorar la precisión de la calibración de la impresora 3D delta. Esta sección proporciona una exposición detallada de la arquitectura del sistema, los algoritmos subyacentes y el flujo de trabajo desde la impresión de un objetivo hasta la actualización del firmware de la máquina.

## Arquitectura del Sistema y Flujo de Trabajo

El sistema propuesto está diseñado como un proceso de bucle cerrado que refina iterativamente los parámetros cinemáticos de la impresora. A continuación, se presenta un diagrama de bloques (Figura 1) de alto nivel del flujo de trabajo, que describe las etapas clave:

1. Impresión del objetivo de calibración: el proceso comienza con la impresora produciendo un artefacto físico especialmente diseñado en su plataforma de construcción.
2. Captura de Imagen: se captura una única imagen estática de la impresión completa en la cama utilizando una cámara monocular.
3. Procesamiento de imágenes y extracción de características: la imagen capturada se somete a una serie de pasos de procesamiento para aislar el objetivo y medir con precisión las posiciones 3D de sus características clave.
4. Cálculo del error: las posiciones medidas se comparan con las posiciones ideales, definidas digitalmente desde el modelo CAD, para calcular un vector de error multidimensional.
5. Optimización de parámetros cinemáticos: un modelo matemático utiliza el vector de error para calcular los ajustes necesarios en los parámetros cinemáticos de la impresora.
6. Actualización del firmware: Los parámetros corregidos se envían al firmware de la impresora,



completando una iteración del bucle de calibración. El proceso puede repetirse hasta que el error medido caiga por debajo de una tolerancia especificada.

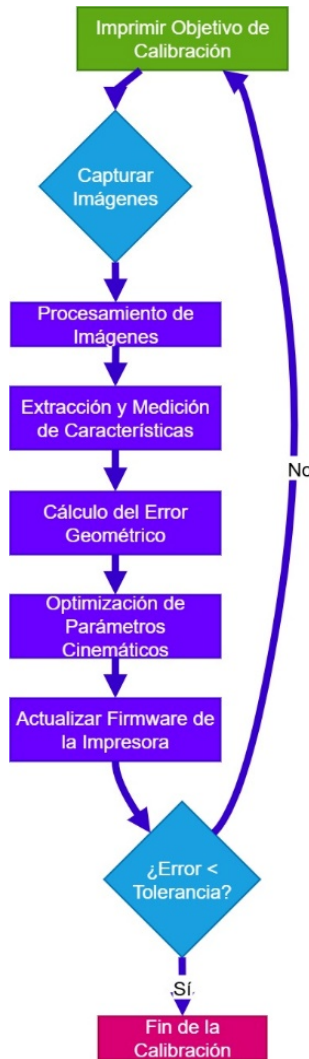


Figura 1. Metodología de calibración propuesta para impresoras delta  
Fuente: Autoría propia.

### Requisitos de Hardware

El sistema está diseñado intencionalmente para ser accesible, basándose en hardware fácilmente disponible y de bajo costo.

- Impresora 3D Delta: cualquier impresora delta que ejecute un firmware que permita la actualización remota de parámetros cinemáticos (por ejemplo, Marlin, Klipper, RepRapFirmware) es compatible.
- Cámara: una cámara digital estándar, como una webcam USB o la cámara de un teléfono inteligente, es suficiente. Una resolución más alta es beneficiosa para la precisión, pero no estrictamente necesaria para que el método funcione.
- Iluminación: para garantizar imágenes consistentes y de alta calidad, se recomienda una iluminación controlada. Una configuración simple, como un anillo de luz LED difusa montado alrededor de la cámara o la iluminación ambiental de la habitación, puede ser eficaz para minimizar sombras y deslumbramientos que podrían interferir con el análisis de la imagen (Clijsters *et al.*, 2014).



## El Objeto de Calibración Delta

El objetivo de calibración físico (Figura 2) es un componente crítico del sistema. Su diseño está optimizado para proporcionar información geométrica clara e inequívoca para todos los parámetros delta clave en una sola impresión. El objetivo incorpora varias características:

- **Marcadores fiduciales:** el objetivo está poblado con varios marcadores ArUco en ubicaciones precisamente conocidas. Los marcadores ArUco son marcadores fiduciales cuadrados con un patrón binario único que permite una detección robusta y rápida. Su principal ventaja es que, a partir de las cuatro esquinas detectadas de un solo marcador, es posible calcular su pose 3D completa (posición y orientación) en relación con la cámara, siempre que la cámara esté calibrada. Esto transforma el problema de la medición de imágenes 2D al análisis espacial 3D (Kalaitzakis *et al.*, 2020).
- **Disposición geométrica:** los marcadores están estratégicamente colocados en un patrón de estrella, correspondiente a las torres de la impresora. Esta disposición es ideal para:
  - Medir las posiciones de los marcadores para ser utilizada en corregir el DELTA\_DIAGONAL\_ROD.
  - Medir los ángulos y posiciones relativas de los marcadores para detectar sesgos o distorsiones trapezoidales, utilizado para corregir DELTA\_TOWER\_ANGLE\_TRIM.

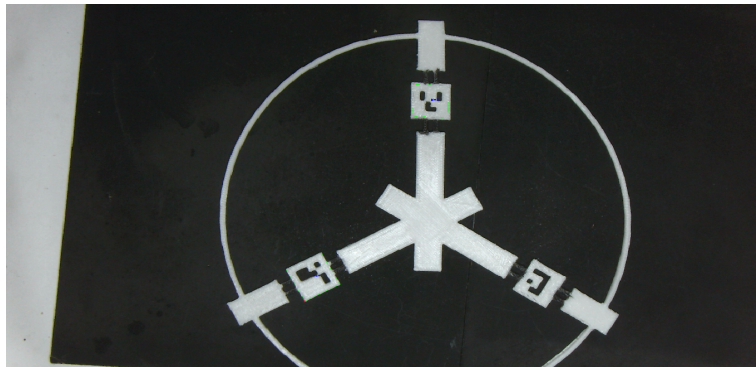


Figura 2. Imagen de pieza de calibración.  
Fuente: Autoría propia.

## Adquisición y preprocesamiento de imágenes de visión artificial

Las imágenes sin procesar de una cámara no son adecuadas para la metrología de alta precisión sin un procesamiento inicial. Esta subsección detalla los dos pasos cruciales necesarios para preparar la imagen para la extracción de características: la calibración de la cámara y la segmentación del objetivo.

El primer y más crítico paso es crear un modelo matemático de la propia cámara para corregir las distorsiones ópticas introducidas por su lente. Las lentes de bajo costo, como las que se encuentran en las webcams, a menudo exhiben una distorsión radial significativa (un efecto de "ojo de pez" o "barril") y distorsión tangencial, que hacen que las líneas rectas en el mundo real aparezcan curvas en la imagen (Heikkila & Silven, 1997; Z. Zhang, 2000). No corregir estas distorsiones introduciría errores de medición significativos, en la Figura 3 se puede ver esta distorsión en una toma.

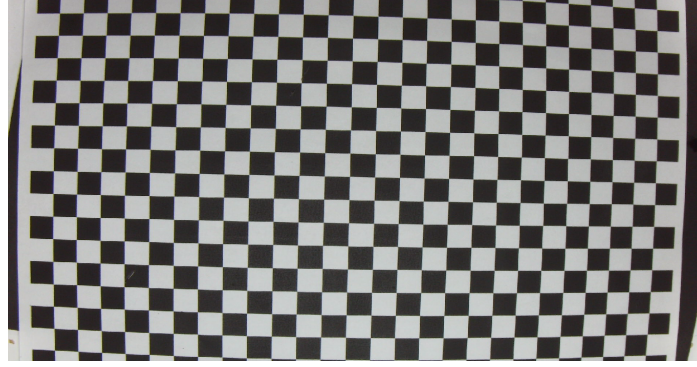


Figura 3. Toma de la cámara con distorsión ojo de pez.  
 Fuente: Autoría propia.

El proceso de calibración implica tomar varias fotografías de un patrón de calibración conocido, típicamente un tablero de ajedrez, desde diferentes ángulos y distancias. La biblioteca OpenCV proporciona funciones robustas para esta tarea, como `cv2.fisheye.calibrate` (O'reilly *et al.*, 2021). Esta función analiza las imágenes del tablero de ajedrez para calcular dos conjuntos clave de parámetros:

- Matriz de la cámara: Esta matriz de  $3 \times 3$  denotada en la ecuación ( 2 ), a menudo denotada como  $K$ , contiene los parámetros intrínsecos de la cámara, incluidas las distancias focales ( $f_x, f_y$ ) y el centro óptico ( $c_x, c_y$ ). Define cómo se proyectan los puntos 3D en el plano de la imagen 2D,  $X_c, Y_c, Z_c$ .
- Coeficientes de distorsión: Este vector contiene los parámetros ( $k_1, k_2, p_1, p_2, \dots$ ) que modelan la distorsión radial y tangencial de la lente, esta corrección se puede ver en la ecuación ( 3 ), la cual da las coordenadas de los píxeles corregidas.

$$s \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = K \begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} \quad (2)$$

$$x_{\text{corregido}} = x(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) + [2p_1 xy + p_2(r^2 + 2x^2)] \quad (3)$$

$$y_{\text{corregido}} = y(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) + [p_1(r^2 + 2y^2) + 2p_2 xy]$$

Una vez que se conocen estos parámetros, cualquier imagen futura capturada por esa cámara puede ser "rectificada" (undistorted) utilizando la función `cv2.undistort`. Este paso de preprocesamiento garantiza que la imagen utilizada para la medición sea una representación geoméricamente de la escena lo más precisa posible, lo cual es fundamental para la precisión de todo el marco, en la Figura 4 se puede ver una toma de cámara con ojo de pez compensado.

Adicionalmente, este patrón de ajedrez permite conocer la escala de los píxeles en milímetros, ecuación ( 4 ), pues al darle el tamaño de cada cuadro en milímetros, openCV puede calcular la escala del píxel al saber cuántos píxeles caben entre la distancia entre vértices de los cuadrados.

$$R_{\frac{\text{px}}{\text{mm}}} = \frac{L_{\text{mm}}}{L_{\text{px}}} \quad (4)$$

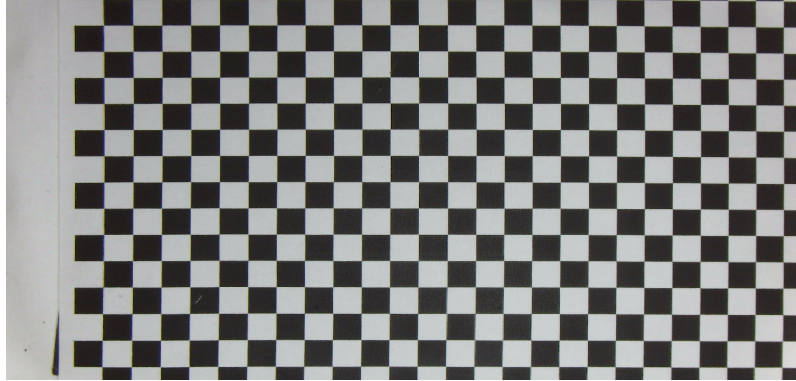


Figura 4. Toma de la cámara con ojo de pez compensado.  
Fuente: Autoría propia.

Después de rectificar la imagen, el siguiente paso es aislar el objetivo de calibración impreso del fondo. Esta es una tarea de segmentación. Si bien la umbralización global simple puede funcionar en condiciones ideales de iluminación uniforme, es notoriamente poco fiable en condiciones del mundo real donde las sombras, los reflejos o la iluminación desigual son comunes (Hammouda *et al.*, 2024). Un único valor de umbral a menudo es insuficiente para separar limpiamente el primer plano del fondo en toda la imagen. Podemos notar esto en la Figura 2, que se puede ver cómo hay cambios en los brillos del modelo, incluso en el fondo, aunque este esté pintado en negro mate.

Para superar esto, el sistema emplea umbralización adaptativa, una técnica que es altamente robusta a las condiciones de iluminación variables. En lugar de utilizar un umbral global, la umbralización adaptativa calcula un umbral diferente para cada píxel en función de los valores de intensidad en su vecindad local (Hammouda *et al.*, 2024; Sauvola & Pietikäinen, 2000). La suposición subyacente es que es más probable que pequeñas regiones de una imagen tengan una iluminación uniforme. La función de OpenCV `cv2.adaptiveThreshold` implementa este método. Dos enfoques comunes son:

- **ADAPTIVE\_THRESH\_MEAN\_C**: el umbral para un píxel es la media de los valores de intensidad en su bloque de vecindad, menos una constante  $C$ , en la ecuación ( 5 ) podemos ver la ecuación de esta, donde  $I(x, Y)$  es la intensidad del píxel, mientras que  $N \times N$  es el tamaño de ventana de la vecindad (Hammouda *et al.*, 2024).

$$T(x, y) = \frac{1}{N \times N} \sum_{i=-\frac{N}{2}}^{\frac{N}{2}} \sum_{j=-\frac{N}{2}}^{\frac{N}{2}} I(x + i, y + j) + C \quad (5)$$

- **ADAPTIVE\_THRESH\_GAUSSIAN\_C**: el umbral, dado por la ecuación ( 6 ), es una suma ponderada de los valores de los píxeles vecinos (utilizando una ventana gaussiana, representada por  $\omega(i, j)$ ), menos la constante  $C$ . Esto da más peso a los píxeles más cercanos al centro del bloque, haciéndolo más robusto al ruido (Hammouda *et al.*, 2024).

$$T(x, y) = \sum_{i=-\frac{N}{2}}^{\frac{N}{2}} \sum_{j=-\frac{N}{2}}^{\frac{N}{2}} \omega(i, j) I(x + i, y + j) + C \quad (6)$$

El resultado de este paso es una imagen binaria limpia y de alto contraste donde el objetivo impreso aparece como píxeles blancos y el fondo es negro (o viceversa), lista para la detección de características.

## Medición de características de alta precisión

Con una imagen binaria limpia, el sistema ahora puede extraer mediciones precisas. El uso de marcadores ArUco proporciona un método particularmente poderoso para esto.

### *Detección de Marcadores ArUco y Estimación de Pose*

La elección de usar marcadores ArUco es una decisión metodológica clave que eleva el sistema de una simple medición 2D a una estimación de pose 3D completa. Esto es especialmente importante, ya que el centro de la figura de calibración está alienado con el centro del ArUco.

El proceso de detección y análisis por visión artificial, manejado por la biblioteca cv2.aruco, procede de la siguiente manera (Kalaitzakis *et al.*, 2020):

1. Detección de Candidatos: el algoritmo primero escanea la imagen en grises, a la cual se le reduce la definición a 720p, en busca de contornos de forma cuadrada, que son candidatos potenciales a marcadores.
2. Extracción e identificación de patrones: para cada cuadrado candidato, se aplica una transformación de perspectiva para deformarlo a su vista canónica (plana, frontal). El algoritmo luego lee el patrón binario interno.
3. Coincidencia con diccionario y corrección de errores: este patrón binario se compara con un diccionario conocido de marcadores ArUco. Si se encuentra una coincidencia (potencialmente después de aplicar corrección de errores para tener en cuenta defectos de impresión menores), el marcador se identifica con éxito y se devuelven su ID único y las coordenadas de píxeles de sus cuatro esquinas, las cuales son escaladas para corresponder a la definición 4K.

El paso más crítico sigue a la detección. Usando la función cv2.aruco.estimatePoseSingleMarkers, el sistema puede calcular la pose 3D de cada marcador detectado. Esta función toma tres entradas: las coordenadas de píxeles 2D de las esquinas del marcador, el tamaño conocido del marcador en el mundo real (por ejemplo, 20 mm x 20 mm), y la matriz de la cámara y los coeficientes de distorsión obtenidos del paso de calibración de la cámara. Luego resuelve el problema de Perspectiva-n-Puntos (PnP), el cual se muestra en la ecuación (7), para encontrar la transformación 3D única que mapea el modelo 3D del marcador a su proyección de imagen 2D, esta representación del mundo se da en las coordenadas  $X_W, Y_W, Z_W$ . La salida es un par de vectores para cada marcador (Kalaitzakis *et al.*, 2020):

- Vector de Rotación (rvec): un vector 3x1 que representa la orientación del marcador en el espacio 3D, que está representado por  $R$  en la ecuación (7) (Xavier *et al.*, 2017).
- Vector de Traslación (tvec): un vector 3x1 que representa la posición (X,Y,Z) del centro del marcador en el sistema de coordenadas de la cámara, que está representado por  $R$  en la ecuación (7).

$$s \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = K \left( R \begin{bmatrix} X_W \\ Y_W \\ Z_W \end{bmatrix} + t \right) \quad (7)$$

El resto de variables en la ecuación (7) son los mismos que se encuentran en el modelo de cámara pinole.

Al repetir estas operaciones en los 3 marcadores fiduciales del modelo de calibración, se obtiene la representación 3D del sistema, al tener que usar vista cenital, no se requiere la calibración espacial o se le da importancia a esto, es en el eje Y donde está nuestro principal interés, ya que el eje Y de los marcadores marca el eje de las barras de calibración, de esta forma se puede, posteriormente, revisar el tamaño de la barra o calcular el ángulo entre barras.

### *Método complementario: análisis de perfil de línea*

Para medir características lineales simples o distancias entre puntos donde no hay marcadores en visión artificial, se puede utilizar un algoritmo de análisis de perfil de línea como técnica complementaria. Este



método logra una precisión al píxel al analizar los gradientes de intensidad (Tu *et al.*, 2025). El proceso implica:

1. Definir una trayectoria: se dibuja por medio de openCV una línea o curva a través de la característica de interés en la imagen, que corresponde al eje Y del marcador fiducial.
2. Graficar la intensidad: se extraen y grafican los valores de intensidad de los píxeles a lo largo de esta trayectoria, formando una señal 1D o "perfil de línea".
3. Detección de bordes: un borde en la imagen corresponde a un cambio brusco en este perfil de intensidad. Al calcular la primera o segunda derivada del perfil, ecuación ( 8 ) y ( 9 ) respectivamente, estos cambios se pueden localizar con alta precisión, a menudo interpolando entre píxeles para encontrar la ubicación del borde con resolución píxel (Xavier *et al.*, 2017). Esto se puede utilizar para medir con precisión el ancho de las líneas impresas o la distancia entre características.

$$\text{Borde} = \arg \max_x \left| \frac{dP(x)}{dx} \right| \quad (8)$$

$$\frac{d^2P(x)}{dx^2} = 0 \quad (9)$$

### Algoritmo de optimización de parámetros cinemáticos

La etapa final del sistema de visión artificial es traducir los errores geométricos medidos en acciones correctivas para el firmware de la impresora. Este es un problema inverso: dados los errores de salida observados, ¿cuáles fueron los errores de los parámetros de entrada que los causaron?

#### Modelo de corrección matemática

La base de la corrección matemática se basa en la corrección de la longitud de la barra diagonal (DIAGONAL\_ROD, D), este proceso nos permite hacer que las distancias calculadas por el modelo cinemático se relacionen con las dimensiones físicas reales.

Para lo cual se necesita saber el punto donde la altura del carro corresponde a la posición, ya que su movimiento está directamente relacionado con el del efector, en la ecuación ( 10 ) tenemos a  $h^*$  que es la altura respecto al efector cuando este está al centro de la estructura,  $D_0^2$  es el cuadrado de la longitud de la barra diagonal dada al controlador y  $R^2$  es el cuadrado del radio horizontal, es decir, la distancia conformada del centro del efectos a el inicio de los brazos.

$$h^* = \sqrt{D_0^2 - R^2} \quad (10)$$

Con la anterior ecuación como base, podemos obtener  $Dh$  el cual es la distancia que se mueve el efector en el plano horizontal, el cual está definido por la ecuación ( 11 ), en esta destacamos a  $L^*$  que es la distancia en el plano horizontal que se le ordenó moverse, para este punto se denota que la distancia de la barra diagonal tiene un impacto muy significativo en las coordenadas.

$$Dh = \sqrt{D_0^2 - (R - L^*)^2} - h^* \quad (11)$$

Por último, tenemos a la ecuación ( 12 ), la cual es la ecuación que calcula cuál sería la longitud  $L$  del movimiento en el plano horizontal real, para esto se toma lo que sería la distancia de barra diagonal real  $D$ , y es esta distancia la que debemos despejar en caso de que  $L$  y  $L^*$  no sean iguales. Para muestra, en la ecuación ( 13 ) tenemos cómo serían la ecuación donde estuvieran desplegadas todas las ecuaciones juntas (Salmerón Valdivieso, 2017).

$$L = R - \sqrt{D^2 - (\sqrt{D^2 - R^2} + Dh)^2} - h^* \quad (12)$$



$$L = R - \sqrt{D^2 - \left( \sqrt{D^2 - R^2} + \sqrt{D_0^2 - (R - L^*)^2} - \sqrt{D_0^2 - R^2} \right)^2} \quad (13)$$

Posterior a haber obtenido la nueva distancia de barra diagonal, se debe de repetir por cada uno de los brazos de la impresora, y obtener el promedio de los 3, estos serán datos que requiere el software de control.

Debido a la linealización de un sistema no lineal, un solo cálculo puede no ser suficiente para lograr una calibración perfecta. Por lo tanto, el marco está diseñado para ser iterativo. Después de que se calcula y aplica el primer conjunto de correcciones al firmware, el proceso puede repetirse: se imprime un nuevo objetivo, se captura una nueva imagen y se mide un nuevo vector de error, más pequeño. Este proceso iterativo continúa hasta que la magnitud del vector de error cae por debajo de un umbral de tolerancia predefinido, asegurando que la calibración converja a un estado de alta precisión.

### Validación experimental y análisis de rendimiento

Para la validación del sistema de calibración propuesto, se llevó a cabo un proceso de medición, tanto manual como automatizada, esto para poder ver la distancia tanto registrada como real. Con el objetivo de comparar, cuantitativamente, el rendimiento del software de calibración, en la Figura 5 A se pueden ver las partes de la puesta en marcha de la medición y en la Figura 5 B la impresora delta mini Kossel que participó en los experimentos.

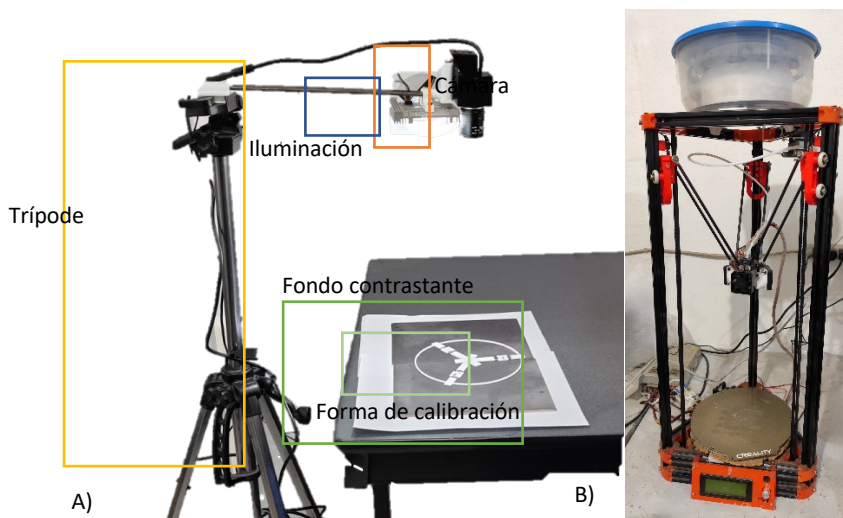


Figura 5. A) Puesta en marcha del sistema de medición B) Impresora delta mini Kossel usada para experimentos.  
Fuente: Autoría propia.

### Software de calibración: interfaz y flujo de trabajo

Adicionalmente, se desarrolló un programa en Python (Figura 7) para facilitar el uso del sistema y la exportación de datos. El programa, compuesto por múltiples funciones, se ilustra en la Figura 6 y su flujo de trabajo se divide en los siguientes pasos, los cuales están resumidos en la Figura 6.

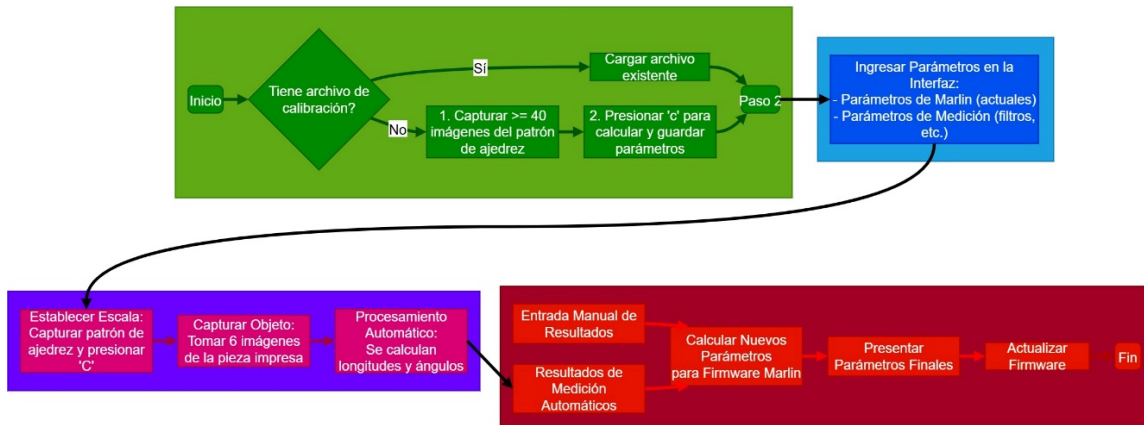


Figura 6. Diagrama de uso del programa.  
Fuente: Autoría propia.

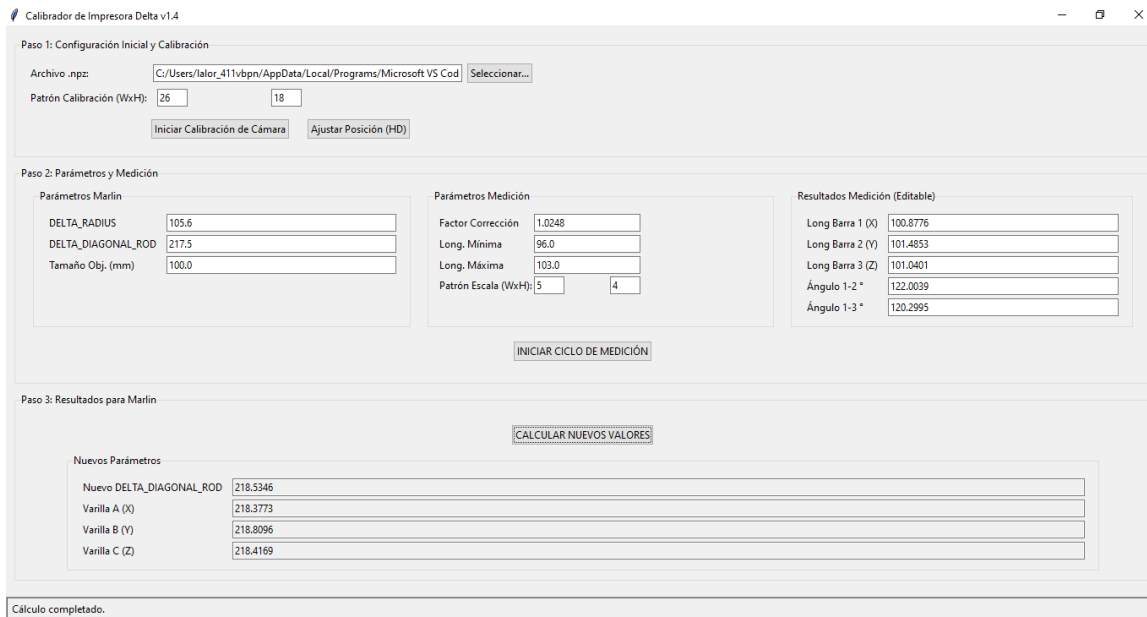


Figura 7. Programa para calibración delta.  
Fuente: Autoría propia.

### Paso 1: Calibración de la cámara

La primera sección se dedica a la calibración de la cámara. Aquí, el usuario puede cargar un archivo de calibración existente o generar uno nuevo. Para ello, se utiliza un patrón de ajedrez con cuadros de 10 mm por lado, que su cantidad de filas y columnas puede ser modificado. El proceso consiste en:

- Capturar al menos 40 imágenes del patrón desde distintos ángulos y posiciones utilizando la resolución 4K de la cámara (cada imagen se toma presionando la tecla 's').
- Para calibrar se debe presionar la tecla 'c', el software utilizará estas imágenes para calcular los parámetros intrínsecos y los coeficientes de distorsión de la lente.
- El archivo de calibración resultante se guarda automáticamente y se carga para su uso posterior.

Esta sección también incluye una herramienta de visualización en HD que ofrece una mayor tasa de fotogramas por segundo para facilitar el enfoque manual de la lente de forma precisa y simple.

### Paso 2: Parámetros y Medición

Esta sección se divide en tres áreas principales para la entrada de datos y la visualización de resultados:



- Parámetros de Marlin: se ingresan los parámetros cinemáticos actuales del firmware de la impresora, así como la longitud esperada de las barras del objeto de calibración impreso.
- Parámetros de Medición: aquí se configuran un factor de corrección para compensar desviaciones sistemáticas en la medición, un filtro de longitud (mínima y máxima) para descartar automáticamente mediciones anómalas causadas por reflejos o errores de umbralización y las medidas de filas y columnas para el patrón de ajedrez que se usará como escala.
- Resultados de Medición: este panel muestra los resultados numéricos (longitudes y ángulos) obtenidos por el algoritmo de visión artificial tras analizar las imágenes.

La fase principal del proceso de medición (Figura 8 A) tiene los siguientes pasos:

- Establecer la Escala: primero, se coloca el patrón de ajedrez con cuadros de 10 mm en el área de trabajo y se presiona la tecla 'C'. El programa utiliza esta imagen de referencia para calcular la escala, es decir, la relación exacta de milímetros por píxel (Figura 8 B).
- Capturar el Objeto: luego, se toman seis imágenes del objeto de calibración impreso con la letra 'm'. Se recomienda girar la pieza 60 grados entre cada captura para promediar las mediciones y minimizar el impacto de posibles errores residuales de la calibración o la perspectiva.
- Procesar: al tomar la sexta imagen, la ventana de captura se cierra y el software procesa automáticamente los datos, mostrando los resultados en la sección correspondiente de la interfaz.

En la Figura 8 se puede ver una representación visual de lo que el sistema está midiendo para obtener las mediciones de las barras, como se puede ver, no es del todo perfecto, de ahí la necesidad del promedio

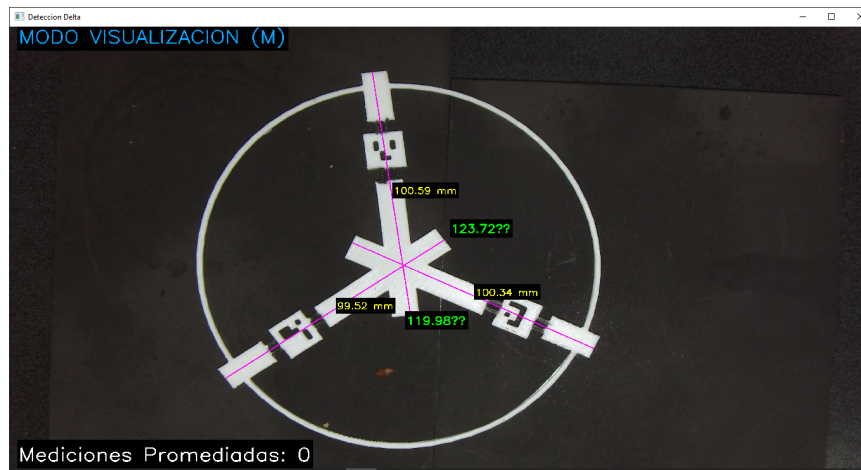


Figura 8. Visualización de la medición del sistema de medición.  
Fuente: Autoría propia.

### Paso 3: Cálculo y Exportación de Resultados

La sección final presenta los nuevos parámetros cinemáticos calculados, listos para ser actualizados en el firmware Marlin. El programa también ofrece flexibilidad, permitiendo la entrada manual de los datos de medición para que el usuario pueda utilizar sus propias herramientas si lo desea.

### Configuración Experimental

Todos los experimentos se realizaron utilizando un entorno de hardware y software consistente para garantizar una comparación justa.

- Impresora 3D: es una impresora delta mini Kossel personalizada con una edad de 10 años, esta se le ha mejorado la precisión con bandas de alma de acero y una extrusora bimetálica clon de una bambu lab X1 carbon, con una cama magnética PEI. El material usado es PETG de color blanco, por problemas con la extrusión se optó por este material, ya que el PLA tuvo algunos problemas de atasco.
- Sistema de Visión: la cámara usada es una cámara USB ELP 4K, 3840 px x 2160 px, basada en el chip IMX317 de Sony, pensada para videovigilancia, pero con control manual de longitud focal, enfoque e iris. Esta se colocó fija en vista cenital a aproximadamente 40 cm a la pieza a medir, ocupando todo el



campo visible posible sin afectar las barras, esto con el fin de mejorar la relación píxel-mm. La iluminación la dio una barra de LEDs (diodos emisores de luz) personalizada con un filtro difusor para aplanar la luz, comportándose como una fuente amplia, para evitar reflejos lo más posible se puso en un ángulo tal que los redujera.

- **Software:** el software de control y análisis se desarrolló en Python 3.9, utilizando la biblioteca OpenCV (versión 4.7) para todas las tareas de visión por computadora, incluida la calibración de la cámara, la umbralización adaptativa y la detección y estimación de pose de marcadores ArUco. La biblioteca NumPy se utilizó para todos los cálculos numéricos.

## Metodología de Comparación

Para la comparación se hizo un ciclo de calibración con el sistema basado en visión artificial y medición manual con un vernier marca Mitutoyo, con una precisión de 0.01 mm. Para llevar a cabo las diferentes mediciones, se imprimió un patrón de calibración tipo estrella con 3 ArUco con los id 1, 2 y 3 para marcar cada una de las torres de la impresora (Figura 2), a propósito del experimento se imprimió a 2 mm. Esta se imprimió con la cama PEI a 70° C, y una temperatura de boquilla de 230° C, se esperó a un enfriamiento completo y paulatino para evitar deformaciones térmicas.

### Métodos de Calibración Bajo Prueba

1. **Línea de Base (Calibración Manual):** Para este método se tomó a un operador experimentado con en la calibración de impresoras 3D delta, se tomó el patrón impreso por la impresora y se realizó la medición usando la sección de medidas externas del vernier, usando la zona más gruesa del vernier en la parte más central del patrón para obtener las mejores mediciones.
2. **Método Propuesto (Calibración Basada en Visión):** Se ejecutó el flujo de trabajo semiautomatizado descrito al inicio de la sección 4. Esto implicó imprimir el objetivo de calibración, ejecutar el script de Python, calibrar o cargar el archivo de calibración de la cámara y capturar 6 imágenes para calcular y aplicar las correcciones cinemáticas. Para este experimento, se realizó una sola iteración del bucle basado en visión.

## Resultados Cuantitativos

Los datos recopilados de los experimentos comparativos se resumen en la Tabla 2. Los valores representan el error absoluto y el relativo de las mediciones del método óptico, usándose como base el método por vernier.

**Tabla 2: Comparación Cuantitativa de Métodos de Calibración**

| Método               | Medición vernier | Medición óptica | Error Absoluto | Error Relativo |
|----------------------|------------------|-----------------|----------------|----------------|
| <b>Torre 1 (mm)</b>  | 99.88            | 100.8776        | 0.9976         | 1.00%          |
| <b>Torre 2 (mm)</b>  | 100.64           | 101.4853        | 0.8453         | 0.84%          |
| <b>Torre 3 (mm)</b>  | 100.64           | 101.0401        | 0.4001         | 0.40%          |
| <b>Promedio (mm)</b> | 100.3866667      | 101.1343333     | 0.747666667    | 0.75%          |

Para la comparativa de rendimiento, se midieron los tiempos del proceso, registrando 2 minutos para la calibración manual y menos de 1 minuto para el método óptico automatizado. Adicionalmente, se calculó la escala de la imagen, obteniendo una relación de 0.07226 mm por píxel.



## Discusión

Los resultados experimentales presentados en la sección anterior proporcionan una sólida base cuantitativa para evaluar el marco de calibración basado en visión propuesto. Esta sección ofrece una interpretación de estos resultados, discute las fortalezas y limitaciones inherentes del enfoque, analiza las posibles fuentes de error y considera las implicaciones más amplias para la comunidad de manufactura aditiva.

## Interpretación de Resultados

El rendimiento del sistema de visión artificial tuvo una dualidad de resultados. Por un lado, demostró ser muy fácil de utilizar por el operario, requiriendo solamente del uso de pocos botones y un enfoque simple, además de reducir en un 50% el tiempo necesario para la calibración. Por otro lado, presentó un error relativo promedio del 0.75% tomando como base la medición manual. Si bien este nivel de precisión puede requerir mejoras para ciertas aplicaciones críticas, es coherente con las limitaciones que presenta el sistema, como se analiza a continuación

La principal fuente de error del sistema de visión artificial es el propio sistema de procesamiento de imágenes. El algoritmo de umbralización que es para detectar los bordes con más precisión usa una matriz de vecindad de 51x51 píxeles, siendo los de más significancia los más cercanos. Considerando que la escala de la imagen en este caso es de 0.07226 mm por píxel, el tamaño del error de los píxeles más significativos del kernel de la umbralización adaptativa, así como la precisión a nivel píxel, da como resultado un error de medición máximo de  $\pm 0.578$  mm, esto considerando que se tiene un error máximo en cada extremo de la línea de medición de las barras. Este error es muy aproximado al error absoluto promedio de 0.747 mm, esto sugiere que la principal limitante es la resolución de la imagen más que un error en la metodología que se ha llevado a cabo. A este error cuantificable se suman los distintos ruidos esporádicos en la cámara que, aunque son muy pocos por el uso del promedio de imágenes, aun así, pueden afectar a las mediciones, y las deficiencias en la calibración que se puedan encontrar, generalmente por el simple hecho de que es casi imposible lograr una calibración perfecta, se suman a la discrepancia medida. Hay que destacar que el error principal del sistema se da en escala de píxeles, donde a más grandes son estos, es decir, a menor definición se tenga, este error se incrementará, por lo que el aumentar de definición de la cámara podría mitigarlos.

En contraste, la facilidad de uso demostró ser excepcional, validando uno de los puntos principales del trabajo. La metodología propuesta resulta ser muy fluida ya que, a excepción del proceso de calibración de la cámara que puede resultar tedioso, pero sólo se realiza una vez, el resto de los procedimientos van guiando al usuario para llevarlos a cabo, reduciendo la subjetividad y la curva de aprendizaje de los métodos manuales. Adicionalmente, el programa incluye herramientas avanzadas, como el poder colocar los valores de tamaño directamente en el programa o el poder especificar un factor de corrección para eliminar errores sistemáticos, permite a los operarios avanzados implementar el sistema como parte de su flujo de trabajo.

## Análisis de Fuentes de Error

Las fuentes de error dentro del propio sistema de visión son principalmente computacionales y ópticas. La precisión final es una combinación de varios factores:

1. Error de Calibración de la Cámara: cualquier inexactitud en el cálculo de la matriz de la cámara y los coeficientes de distorsión se propagará a través de todo el pipeline de medición. Esto se puede minimizar utilizando un objetivo de calibración de alta calidad y tomando un gran número de imágenes desde diversos ángulos durante el proceso de calibración de la cámara.
2. Cuantificación de Píxeles: una imagen digital es una cuadrícula discreta. La ubicación de una característica solo puede determinarse hasta el píxel más cercano, aunque los algoritmos sub-píxel pueden estimar posiciones entre píxeles. El efecto de este error de cuantificación se reduce a medida que aumenta la resolución de la cámara y el tamaño de las características en la imagen.
3. Precisión de Detección de ArUco: la biblioteca cv2.aruco es muy robusta, pero la detección de sus esquinas puede verse afectada por artefactos de impresión como esquinas redondeadas o "ringing". Asegurarse de que el objetivo de calibración se imprima a una velocidad moderada con buena



refrigeración puede mejorar la nitidez de las características y la precisión de la detección.

## Tamaño de kernel de umbralización: Interpretación de Resultados

El rendimiento del sistema de visión artificial tuvo una dualidad de resultados. Por un lado, demostró ser muy fácil de utilizar por el operario, requiriendo solamente del uso de pocos botones y un enfoque simple, además de reducir en un 50 % el tiempo necesario para la calibración. Por otro lado, presentó un error relativo promedio del 0.75 % tomando como base la medición manual. Si bien este nivel de precisión puede requerir mejoras para ciertas aplicaciones críticas, es coherente con las limitaciones que presenta el sistema, como se analiza a continuación

La principal fuente de error del sistema de visión artificial es el propio sistema de procesamiento de imágenes. El algoritmo de umbralización que es para detectar los bordes con más precisión usa una matriz de vecindad de 51x51 píxeles, siendo los de más significancia los más cercanos. Considerando que la escala de la imagen en este caso es de 0.07226 mm por píxel, el tamaño del error de los píxeles más significativos del kernel de la umbralización adaptativa, así como la precisión a nivel píxel, da como resultado un error de medición máximo de  $\pm 0.578$  mm, esto considerando que se tiene un error máximo en cada extremo de la línea de medición de las barras. Este error es muy aproximado al error absoluto promedio de 0.747 mm, esto sugiere que la principal limitante es la resolución de la imagen más que un error en la metodología que se ha llevado a cabo. A este error cuantificable se suman los distintos ruidos esporádicos en la cámara que, aunque son muy pocos por el uso del promedio de imágenes, aun así, pueden afectar a las mediciones, y las deficiencias en la calibración que se puedan encontrar, generalmente por el simple hecho de que es casi imposible lograr una calibración perfecta, se suman a la discrepancia medida. Hay que destacar que el error principal del sistema se da en escala de píxeles, donde a más grandes son estos, es decir, a menor definición se tenga, este error se incrementará, por lo que el aumentar de definición de la cámara podría mitigarlos.

En contraste, la facilidad de uso demostró ser excepcional, validando uno de los puntos principales del trabajo. La metodología propuesta resulta ser muy fluida ya que, a excepción del proceso de calibración de la cámara que puede resultar tedioso, pero sólo se realiza una vez, el resto de los procedimientos van guiando al usuario para llevarlos a cabo, reduciendo la subjetividad y la curva de aprendizaje de los métodos manuales. Adicionalmente, el programa incluye herramientas avanzadas, como el poder colocar los valores de tamaño directamente en el programa o el poder especificar un factor de corrección para eliminar errores sistemáticos, permite a los operarios avanzados implementar el sistema como parte de su flujo de trabajo.

## Análisis de Fuentes de Error

Las fuentes de error dentro del propio sistema de visión son principalmente computacionales y ópticas. La precisión final es una combinación de varios factores:

1. Error de Calibración de la Cámara: cualquier inexactitud en el cálculo de la matriz de la cámara y los coeficientes de distorsión se propagará a través de todo el pipeline de medición. Esto se puede minimizar utilizando un objetivo de calibración de alta calidad y tomando un gran número de imágenes desde diversos ángulos durante el proceso de calibración de la cámara.
2. Cuantificación de Píxeles: una imagen digital es una cuadrícula discreta. La ubicación de una característica solo puede determinarse hasta el píxel más cercano, aunque los algoritmos sub-píxel pueden estimar posiciones entre píxeles. El efecto de este error de cuantificación se reduce a medida que aumenta la resolución de la cámara y el tamaño de las características en la imagen.
3. Precisión de Detección de ArUco: la biblioteca cv2.aruco es muy robusta, pero la detección de sus esquinas puede verse afectada por artefactos de impresión como esquinas redondeadas o "ringing". Asegurarse de que el objetivo de calibración se imprima a una velocidad moderada con buena refrigeración puede mejorar la nitidez de las características y la precisión de la detección.



4. Tamaño de kernel de umbralización: si bien el kernel Gausiano de 51x51 es eficaz para manejar variaciones de iluminación en el fondo, su gran tamaño también puede ser una fuente de error. Al promediar el umbral sobre un área tan extensa, puede "suavizar" la ubicación del borde real del objeto, reduciendo la precisión de su localización. Esta imprecisión, magnificada por la escala de 0.07226 mm/píxel, se suma al error de cuantificación general.

## Conclusión y Direcciones Futuras

Esta investigación demuestra exitosamente que la visión por computadora ofrece una solución eficaz y accesible al complejo desafío de la calibración en impresoras 3D delta. Al reemplazar los ajustes manuales subjetivos y las mediciones mecánicas indirectas por un análisis de imagen objetivo, el marco propuesto cierra la brecha de habilidad y conocimiento técnico que ha limitado la adopción masiva de esta tecnología de impresión de alta velocidad.

La contribución principal de este trabajo es un sistema completo de extremo a extremo que traduce una serie de imágenes de un artefacto impreso en correcciones precisas para los parámetros cinemáticos de la impresora. La metodología, centrada en marcadores fiduciales ArUco y un algoritmo de optimización, permite la extracción de datos 3D a partir de una simple imagen 2D. La validación experimental reveló que, si bien la precisión dimensional no superó a la de un operario experto, el sistema destaca por su excepcional facilidad de uso y una curva de aprendizaje drásticamente reducida, permitiendo que usuarios novatos obtengan resultados consistentes y fiables.

Las implicaciones de este enfoque son significativas. Al simplificar drásticamente el aspecto más difícil de operar una impresora delta, esta tecnología podría hacer que la impresión de alta velocidad y precisión sea más accesible para aficionados, educadores y pequeñas empresas. Para entornos comerciales como las granjas de impresión, una rutina de calibración automatizada podría integrarse en los flujos de trabajo de control de calidad, permitiendo recalibraciones automáticas para aumentar el tiempo de actividad, reducir residuos y garantizar una producción consistente.

De cara al futuro, este trabajo abre varias vías de investigación prometedoras. El marco podría adaptarse a otras cinemáticas complejas (CoreXY, SCARA), o integrarse con modelos de aprendizaje profundo para eliminar la necesidad de un objeto de calibración específico. La evolución definitiva sería fusionar este sistema con técnicas de monitoreo en tiempo real, creando un sistema de control de bucle cerrado que ajuste la calibración durante la impresión para lograr una precisión perfecta de principio a fin, representando un paso clave hacia una manufactura aditiva verdaderamente autónoma.

## Referencias

- Bonev, I. A. (2001). *The True Origins of Parallel Robots*. ParalleMIC Research Group.
- Clavel, R. (1988). DELTA, a fast parallel robot for pick-and-place operations. *Proceedings of the 18th International Symposium on Industrial Robots (ISIR)*.
- Clijsters, S., Craeghs, T., Buls, S., Van Hooreweder, B., & Kruth, J. P. (2014). In situ quality control of the selective laser melting process using a high-speed, real-time camera system. *The International Journal of Advanced Manufacturing Technology*, 75(5–8), 1089–1101.
- Ganitano, G. S., Maruyama, B., & Peterson, G. L. (2025). Accelerated Multiobjective Calibration of Fused Deposition Modeling 3D Printers Using Multitask Bayesian Optimization and Computer Vision. *Advanced Intelligent Systems*, 7(4). <https://doi.org/10.1002/AISY.202400523>
- Hammouda, A. Ben, Frikha, A., Koubaa, S., & Mrad, H. (2024). Defect detection in additive manufacturing using image processing techniques. *Procedia Computer Science*, 232, 2157–2166. <https://doi.org/10.1016/J.PROCS.2024.02.035>
- Heikkila, J., & Silven, O. (1997). A four-step camera calibration procedure with implicit image correction. *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*.



- Kalaitzakis, M., Carroll, S., & Vitzilaios, N. (2020). ArUco-based Localization and Mapping for Autonomous Underwater Vehicles. *2020 IEEE/OES Autonomous Underwater Vehicles Symposium (AUV)*.
- Kang, R., Sang, L., Yang, L., Yang, K., Hao, R., Zhang, H., & Sang, S. (2023). 3D printer vision calibration system based on embedding Sobel bilateral filter in least squares filtering algorithm. *The Visual Computer*, 40(9), 6599–6613. <https://doi.org/10.1007/S00371-023-03187-0>
- Klipper. (2023a). *Delta Calibration*. [https://www.klipper3d.org/Delta\\_Calibrate.html](https://www.klipper3d.org/Delta_Calibrate.html)
- Klipper. (2023b). *Kinematics*. <https://www.klipper3d.org/Kinematics.html>
- Marlin Firmware. (2023). *Delta Kinematics*. <https://marlinfw.org/docs/gcode/G033.html>
- Moser, B. L., Gordon, J. A., Petruska, A. J., Boschetti, G., Miguel, J., & Sousa, C. (2021). Unified Parameterization and Calibration of Serial, Parallel, and Hybrid Manipulators. *Robotics 2021*, 10, pp. 124, 10(4), 124. <https://doi.org/10.3390/ROBOTICS10040124>
- O'reilly, C. S., Elbadawi, M., Desai, N., Gaisford, S., Basit, A. W., & Orlu, M. (2021). Machine learning and machine vision accelerate 3d printed orodispersible film development. *Pharmaceutics*, 13(12), 2187. <https://doi.org/10.3390/PHARMACEUTICS13122187/S1>
- Pierrot, F., & Company, O. (1999). H4: A new family of 4-dof parallel robots. *Advances in Robot Kinematics*, 435–442.
- RepRap. (2023a). *Configuring RepRapFirmware for a Delta printer*. [https://docs.duet3d.com/en/User\\_manual/Machine\\_configuration/Configuration\\_delta](https://docs.duet3d.com/en/User_manual/Machine_configuration/Configuration_delta)
- RepRap. (2023b). *Delta Calibration*. [https://reprap.org/wiki/Delta\\_calibration](https://reprap.org/wiki/Delta_calibration)
- Salmerón Valdivieso. (2017). *Diagonal rod correction for Delta printing XY dimensional calibration*. <https://www.thingiverse.com/thing:1274733>
- Sauvola, J., & Pietikäinen, M. (2000). Adaptive document image binarization. *Pattern Recognition*, 33(2), 225–236.
- Silva, J. M., Wagner, G., Silva, R., Morais, A., Ribeiro, J., Mould, S., Figueiredo, B., Nóbrega, J. M., & Cruz, P. J. S. (2024). Real-Time Precision in 3D Concrete Printing: Controlling Layer Morphology via Machine Vision and Learning Algorithms. *Inventions 2024*, 9, pp. 80, 9(4), 80. <https://doi.org/10.3390/INVENTIONS9040080>
- Taufik, M., & Jain, P. K. (2016). Role of build orientation in layered manufacturing: a review. *International Journal of Manufacturing Technology and Management*, 30(1–2), 154–182.
- Temistocles, M., Salazar, A., Alejandra, A., Lucas, B., Katherine, R., Morillo, B., David, M., & Sasig, S. (2023). Control y monitoreo remoto de una impresora 3D con visión artificial. *Código Científico Revista de Investigación*, 4(2), 959–968. <https://doi.org/10.55813/GAEA/CCRI/V4/N2/266>
- Tu, Y., Zhang, H., Chen, H., Bao, B., Fang, C., Wu, H., Chen, X., Hassan, A., & Boudaoud, H. (2025). Dimensional Accuracy Evaluation of Single-Layer Prints in Direct Ink Writing Based on Machine Vision. *Sensors 2025*, 25, pp. 2543, 25(8), 2543. <https://doi.org/10.3390/S25082543>
- Valiente, D., Fustero-Torre, C., & Toledo, R. (2021). A review on computer vision in additive manufacturing: A focus on dimensional and defect inspection. *Sensors*, 21(16), 5549.
- Wu, Z., Liu, W., & He, Y. (2020). A vision-based real-time monitoring system for FDM 3D printing process. *The International Journal of Advanced Manufacturing Technology*, 107, 269–280.
- Xavier, R. S., Da Silva, B. M. F., & Goncalves, L. M. G. (2017). Accuracy analysis of augmented reality markers for visual mapping and localization. *Proceedings - 13th Workshop of Computer Vision, WVC 2017, 2018-January*, 73–77. <https://doi.org/10.1109/WVC.2017.00020>
- Zhang, D., Caro, S., & Yu, D. (2022). Kinematic Calibration of Parallel Robots Based on L-Infinity Parameter Estimation. *Machines 2022*, 10, pp. 436, 10(6), 436. <https://doi.org/10.3390/MACHINES10060436>
- Zhang, Z. (2000). A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11), 1330–1334.

