

Implementación de un Perceptrón Multicapa en Microcontroladores de Bajo Costo para la Clasificación Cromática Computada en el Borde

Implementation of a Multilayer Perceptron on Low-Cost Microcontrollers for Edge-Computed Chromatic Classification

Machado-Díaz, E.¹; Coto-Fuentes, H.² ✉; Vaquera-González, E.³

Datos de Adscripción:

¹ Machado Díaz, Eduardo
Tecnológico Nacional de México – Instituto Tecnológico Superior de Lerdo.
División de Ingeniería en Sistemas Automotrices. Ciudad Lerdo, Durango.
México
ROR ID: <https://ror.org/00davry38>
ROR ID: <https://ror.org/04bjbt12>
eduardo.md@itslerdo.edu.mx
ORCID: <https://orcid.org/0000-0003-2610-7570>

² ✉ Coto Fuentes, Hesner
Tecnológico Nacional de México – Instituto Tecnológico de La Laguna.
Torreón, Coahuila. México
ROR ID: <https://ror.org/00davry38>
ROR ID: <https://ror.org/00xbw4a70>
Hesner.cf@gmail.com
ORCID: <https://orcid.org/0000-0002-9548-7864>

³ Vaquera González, Efrain
Tecnológico Nacional de México – Instituto Tecnológico Superior de Lerdo.
División de Ingeniería en Sistemas Automotrices. Ciudad Lerdo, Durango.
México
ROR ID: <https://ror.org/00davry38>
ROR ID: <https://ror.org/04bjbt12>
efrain.vg@itslerdo.edu.mx
ORCID: <https://orcid.org/0009-0002-9123-6977>

Resumen - La transición hacia la Industria 4.0 exige sistemas ciberfísicos capaces de procesar información estocástica directamente en el dispositivo, superando las limitaciones del control determinista tradicional. Este artículo presenta el diseño e implementación de una red neuronal artificial, específicamente un perceptrón multicapa (MLP), desplegado en un microcontrolador de arquitectura asimétrica de doble núcleo (RP2040) para realizar tareas de clasificación cromática mediante computación en el borde o Edge AI. La metodología empírica contempló la evaluación de diversos sensores ópticos, seleccionando la tecnología de integración digital directa y filtrado infrarrojo (TCS34725) por su alta estabilidad frente a transductores de conversión de luz a frecuencia. Se generó un conjunto de datos estandarizado y balanceado de 1250 muestras a partir de objetos fabricados en polímero PETG correspondientes a cinco clases de color. La red neuronal propuesta posee una topología ligera de cuatro capas (3-16-16-5) y funciones de activación ReLU y Softmax, resultando en un modelo computacionalmente eficiente de solo 341 parámetros entrenables. El modelo fue entrenado y validado mediante un flujo de aprendizaje supervisado, demostrando una convergencia estable sin presencia de sobreajuste. Los resultados experimentales sobre un conjunto de prueba independiente arrojaron una exactitud global del 99.20% y un puntaje F1 del 99% para las clases de mayor

separabilidad lineal. Se concluye que la implementación de arquitecturas neuronales en sistemas embebidos de bajo costo es altamente viable, permitiendo un procesamiento cognitivo robusto para aplicaciones de control de calidad inteligente en la manufactura moderna.

Palabras Clave - Clasificación cromática, Computación en el borde, Perceptrón multicapa, Redes neuronales artificiales, RP2040, Sistemas embebidos.

Abstract - The transition towards Industry 4.0 demands cyber-physical systems capable of processing stochastic information directly on the device, overcoming the limitations of traditional deterministic control. This article presents the design and implementation of an artificial neural network, specifically a multilayer perceptron (MLP), deployed on an asymmetric dual-core microcontroller (RP2040) to perform chromatic classification tasks through edge computing or Edge AI. The empirical methodology included the evaluation of various optical sensors, selecting direct digital integration and infrared filtering technology (TCS34725) for its high stability compared to light-to-frequency transducers. A standardized and balanced dataset of 1250 samples was generated from objects manufactured in PETG polymer corresponding to five color classes. The proposed neural network features a lightweight four-layer topology (3-16-16-5) along with ReLU and Softmax activation functions, resulting in a computationally efficient model of only 341 trainable parameters. The model was trained and validated through a supervised learning workflow, demonstrating stable convergence without overfitting. Experimental results on an independent test set yielded an overall accuracy of 99.20% and a perfect F1 score for the classes with the highest linear separability. It is concluded that the implementation of neural architectures in low-cost embedded systems is highly viable, enabling robust and decentralized cognitive processing for intelligent quality control applications in modern manufacturing.

Keywords - Artificial neural networks, Chromatic classification, Edge computing, Embedded systems, Multilayer perceptron, RP2040.

I. INTRODUCCIÓN

1.1 Contextualización y la Evolución hacia la Calidad 4.0

En una línea de producción moderna, el color no es un mero atributo estético. Funciona, en realidad, como un indicador físico determinante para evaluar desde el nivel de maduración de un producto agroalimentario hasta el correcto acabado de un componente automotriz. Durante décadas, la automatización clásica resolvió esta inspección mediante sistemas de visión centralizados, costosos y rígidos, o bien, delegando la tarea al ojo humano, un método inevitablemente expuesto a la fatiga y a la subjetividad. Hoy, la transición hacia la denominada Calidad 4.0 rompe con este esquema tradicional. Este nuevo paradigma no busca simplemente automatizar la recolección de fallas, sino descentralizar la inteligencia operativa. La meta actual de la manufactura inteligente consiste en procesar, analizar y decidir directamente en el borde (Edge Computing), es decir, en el mismo instante y lugar donde el sensor interactúa con la pieza física, eliminando por completo la latencia y la dependencia de servidores externos.

La visión artificial aplicada al entorno productivo ha respaldado sólidamente esta evolución. En sectores de alta exigencia como el logístico y el agroindustrial, la clasificación cromática automatizada ha resuelto problemas complejos de variabilidad y ruido óptico.

Un ejemplo claro se observa en el sector alimentario, donde Mendoza de la Cruz con sus colaboradores (2023) demostraron cómo las técnicas de visión computacional y aprendizaje profundo logran estandarizar el control de calidad con una precisión que supera las capacidades de la inspección manual. Asimismo, los esfuerzos recientes por migrar estos procesos hacia plataformas embebidas compactas reflejan una tendencia clara a ganar autonomía en las estaciones de trabajo (Rojas-Hernández et al., 2021).

El desafío técnico contemporáneo, sin embargo, radica en la escala y el costo. Llevar el aprendizaje automático a microcontroladores de ultra bajo consumo, el ecosistema emergente conocido como TinyML (Warden & Situnayake, 2020), exige diseñar algoritmos matemáticamente compactos. El reto es claro: ejecutar tareas de clasificación complejas sin comprometer la velocidad de respuesta en tiempo real ni elevar el presupuesto de la infraestructura industrial.

1.2 Planteamiento del Problema y Limitaciones del Control Tradicional

En las líneas de ensamble automatizadas, los Controladores Lógicos Programables (PLC) constituyen el núcleo operativo estándar debido a su alta confiabilidad en entornos hostiles. Sin embargo, su arquitectura interna de procesamiento no se diseñó para la flexibilidad algorítmica. Un PLC comercial clásico, como el Siemens S7-1200, opera bajo un ciclo de escaneo síncrono (scan cycle) que lee entradas, ejecuta una lógica secuencial y actualiza salidas de forma cíclica (Siemens AG, 2020). Cuando intentamos integrar el procesamiento matricial de una red neuronal artificial dentro de

este flujo determinista, el rendimiento colapsa. Las operaciones matemáticas de punto flotante requeridas para evaluar los pesos y sesgos de un modelo de aprendizaje automático saturan la unidad central de procesamiento del controlador. Esto incrementa el tiempo de ciclo a niveles inaceptables y compromete la sincronización en tiempo real que exige una línea de producción en movimiento constante.

A esta limitación de cómputo se suma la complejidad física intrínseca del sensado óptico. En el piso de producción, el color dista de ser una variable estática o ideal; se comporta, en cambio, como un fenómeno estocástico fuertemente influenciado por el entorno. Factores críticos como las fluctuaciones en la iluminación ambiental, el desgaste de las fuentes de luz artificial y las sutiles variaciones no lineales en la distancia focal entre el objeto y el sensor corrompen la señal original (Sánchez-Torres & Mejía-Álvarez, 2020). La automatización convencional aborda este desafío mediante programación en diagramas de escalera (Ladder Logic), estableciendo umbrales fijos y rígidos para delimitar cada color. Esta estrategia puramente determinista resulta sumamente frágil ante el ruido óptico. Cualquier perturbación mínima en las condiciones lumínicas del laboratorio altera las lecturas de los transductores de bajo costo, lo que descalibra el sistema y provoca constantes falsos rechazos al clasificar polímeros industriales. El problema, por lo tanto, exige una solución capaz de absorber esta no linealidad sin depender de hardware industrial sobredimensionado.

1.3 Estado del Arte y la Brecha del Conocimiento

La ejecución de redes neuronales aplicadas a la clasificación visual dependía de equipos de cómputo de escritorio o de tarjetas de desarrollo robustas como la Raspberry Pi 4. Aunque estas plataformas ofrecen una capacidad de procesamiento incuestionable, su elevado consumo de energía y su costo económico las vuelven inviables cuando se pretende desplegar nodos distribuidos y masivos en un entorno industrial. Algunos antecedentes valiosos intentaron mitigar este problema migrando el aprendizaje automático hacia arquitecturas mucho más compactas. Por ejemplo, Granados y sus colaboradores (2011) lograron alojar un perceptrón multicapa básico en un microcontrolador PIC para identificar colores. Sin embargo, las severas restricciones de memoria volátil y las bajas velocidades de reloj de los semiconductores de aquella época limitaron la precisión del sistema, confinándolo a pruebas experimentales con un número muy reducido de categorías cromáticas.

El panorama técnico cambió drásticamente con la consolidación de *TinyML*, un ecosistema conceptual y tecnológico que optimiza los modelos de aprendizaje automático para que operen con éxito dentro de las restricciones físicas de hardware de un microcontrolador (Warden & Situnayake, 2020). Al unísono de esta evolución algorítmica, los circuitos integrados de bajo costo también dieron un salto cualitativo. Dispositivos contemporáneos de nivel de entrada incorporan ahora arquitecturas de procesamiento multinúcleo, lo que abre una ventana inédita para la automatización en el borde. A pesar de estas ventajas tangibles, la literatura indexada en español todavía exhibe una brecha considerable en el desarrollo de aplicaciones que exploten activamente estas nuevas

capacidades de hardware. Gran parte de las publicaciones regionales se concentran en simulaciones de software o en arquitecturas donde el microcontrolador actúa como un mero puente que envía datos hacia una computadora central, perpetuando la dependencia de una infraestructura externa. Ray (2022) y Sari con sus colaboradores (2023).

Es justamente en este vacío científico donde la presente investigación interviene. El aporte de este artículo no radica únicamente en programar un modelo matemático dentro de un chip económico, sino en resolver de manera explícita el problema de la concurrencia temporal mediante la arquitectura de doble núcleo del microcontrolador RP2040 (Raspberry Pi Ltd., 2024). Para ello, se implementó una estrategia de cómputo en la que un núcleo se encarga exclusivamente de la comunicación síncrona y la adquisición de datos del sensor óptico, mientras el segundo núcleo ejecuta de manera paralela y dedicada las operaciones de punto flotante de la red neuronal. Al aislar las tareas de transducción y de inferencia, se eliminan los retardos que comúnmente degradan el rendimiento en los sistemas mononúcleo convencionales. De este modo, se cubre una brecha crítica en la literatura técnica de automatización, demostrando de forma honesta y reproducible que es posible alcanzar una alta precisión en la clasificación cromática de polímeros sin necesidad de recurrir a plataformas comerciales costosas.

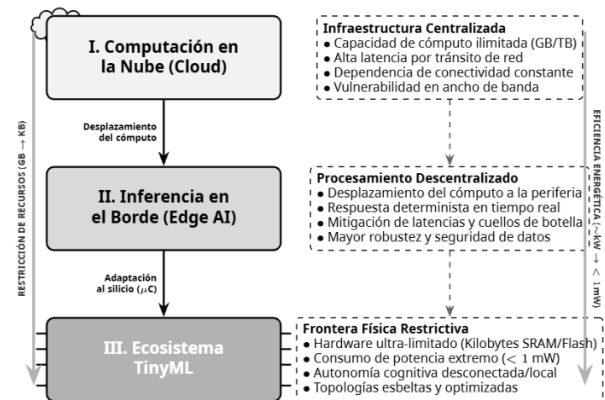
1.4 El paradigma de TinyML y la arquitectura de inferencia en el borde

La computación en la nube transformó la analítica de datos al ofrecer una capacidad de almacenamiento y procesamiento prácticamente ilimitada. Sin embargo, en el contexto de la automatización industrial, transferir datos masivos hacia servidores remotos genera inconvenientes severos de ancho de banda, vulnerabilidades de seguridad y, fundamentalmente, problemas de latencia que impiden una respuesta inmediata en la línea de producción. Para solucionar estas restricciones surgieron las arquitecturas de inferencia en el borde (*Edge Computing*), una aproximación tecnológica que desplaza el cómputo algorítmico directamente hacia la periferia física del sistema, aproximando la toma de decisiones al origen de la captura de los datos (Ray, 2022). Al eliminar los tiempos de tránsito en la red, el procesamiento en el borde habilita respuestas deterministas en tiempo real, transformando nodos sensores pasivos en unidades con autonomía cognitiva local. Dentro de esta tendencia de descentralización, el ecosistema conocido como *TinyML* representa la frontera más restrictiva y eficiente de la Inteligencia Artificial en el Borde. Este paradigma integra algoritmos de aprendizaje automático optimizados dentro de sistemas embebidos basados en microcontroladores, los cuales se caracterizan por operar con consumos de potencia típicamente inferiores al miliwatts (Warden & Situnayake, 2020). La distinción entre un dispositivo embebido convencional y un nodo dotado de *TinyML* radica en la capacidad de este último para ejecutar modelos matemáticos complejos, como las redes neuronales artificiales, de manera local y desconectada. Esta transición tecnológica reduce drásticamente los costos de infraestructura, ya que prescinde de computadoras industriales de gran escala y garantiza la resiliencia del proceso ante fallas

en las redes de comunicación corporativas. El desplazamiento del cómputo algorítmico y las severas restricciones físicas de memoria en el silicio se detallan formalmente en la Figura 1.

Figura 1

Evolución y descentralización de los paradigmas de computación hacia el ecosistema *TinyML* y su modelo de optimización de recursos. Fuente: Elaboración propia bajo el enfoque de Sari et al. (2023).



No obstante, el despliegue de modelos predictivos en microcontroladores impone un severo desafío de diseño debido a las drásticas restricciones de hardware. Mientras que un entorno de desarrollo en la nube dispone de gigabytes de memoria, un microcontrolador estándar opera con recursos medidos en kilobytes tanto para el almacenamiento persistente (memoria Flash) como para la memoria de ejecución dinámica (SRAM). Este escenario obliga a replantear la arquitectura de los modelos, transitando desde estructuras masivas hacia topologías esbeltas y optimizadas numéricamente.

La literatura técnica documenta que el éxito de la inferencia en este ecosistema no depende de emular la capacidad de supercómputo, sino de adaptar con precisión la carga matemática del algoritmo a los límites de silicio del hardware (Sari et al., 2023). De este modo, *TinyML* se consolida no solo como una solución de ingeniería de software, sino como un modelo de optimización que equilibra con honestidad la precisión de la inferencia frente al costo energético y económico del sistema.

1.5 Fundamentos matemáticos del Perceptrón Multicapa enfocado a la inferencia

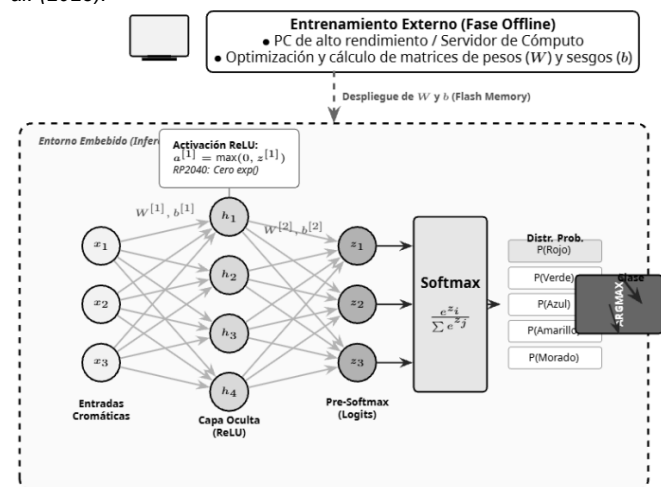
El Perceptrón Multicapa constituye una de las arquitecturas fundacionales del aprendizaje profundo debido a su capacidad demostrada para aproximar funciones no lineales complejas. En el contexto de este estudio, donde el entrenamiento del modelo se realiza de forma previa y externa en un equipo de cómputo de alta capacidad, el microcontrolador asume exclusivamente la fase de explotación o propagación hacia adelante, conocida formalmente como *forward propagation*. Desde una perspectiva algebraico-matricial, este proceso consiste en un flujo secuencial de transformaciones lineales y no lineales distribuidas a lo largo de capas interconectadas. Cada neurona en una capa determinada recibe un vector de entradas proveniente de la etapa anterior, el cual se somete a un producto punto con una matriz de pesos sinápticos que representan el conocimiento numérico adquirido por la red

durante su entrenamiento. A este resultado parcial se le adiciona un vector de sesgo o *bias*, cuya función matemática es desplazar la línea de regresión para permitir un óptimo ajuste a los datos empíricos (Goodfellow et al., 2016).

La simple acumulación de operaciones lineales, sin embargo, resultaría incapaz de resolver problemas de clasificación donde las fronteras de decisión son irregulares, tal como sucede con las coordenadas cromáticas afectadas por perturbaciones ópticas. Por esta razón, la suma ponderada de cada neurona se proyecta a través de una función de activación no lineal. En las capas ocultas de la arquitectura propuesta se seleccionó la Unidad Lineal Rectificada, conocida comúnmente como ReLU por sus siglas en inglés (Haykin, 2009). Esta función introduce una no linealidad de manera muy eficiente, ya que devuelve un valor de cero para cualquier entrada negativa y mantiene el valor original si la entrada es estrictamente positiva. Desde el punto de vista del cómputo embebido, ReLU ofrece una ventaja crítica sobre funciones tradicionales como la sigmoide o la tangente hiperbólica: al no requerir el cálculo de funciones exponenciales densas, reduce de forma notable el uso de ciclos de reloj del procesador RP2040, acelerando la inferencia local sin comprometer la memoria del sistema. La representación esquemática de este apartado se muestra en la Figura 2.

Figura 2

Representación esquemática del flujo algebraico-matricial del Perceptrón Multicapa (MLP) embebida en el microcontrolador RP2040. Fuente: Elaboración propia basada en Goodfellow et al. (2016) y Sari et al. (2023).



El segmento final del procesamiento ocurre en la capa de salida, encargada de consolidar los resultados numéricos en etiquetas lógicas legibles para el entorno de automatización. Para resolver un problema de clasificación multiclase donde las categorías son mutuamente excluyentes, la red utiliza la función de activación Softmax (Bishop, 2006).

Esta función toma los puntajes brutos generados por la última capa oculta y los normaliza, transformándolos en una distribución de probabilidad continua que suma exactamente uno. Cada una de las cinco salidas del perceptrón representa la probabilidad matemática de que la muestra de polímero evaluada pertenezca a un color específico (rojo, verde, azul, amarillo o morado). El microcontrolador finalmente ejecuta un

operador de argumento máximo para seleccionar la clase con el porcentaje de certeza más elevado. Este diseño matemático cerrado permite que un hardware de bajo costo ejecute una clasificación robusta basada en fundamentos probabilísticos, garantizando la repetibilidad del diagnóstico en la periferia de la red.

1.6 Teoría de la transducción óptica y la cuantificación del espacio de color

En el ámbito de la automatización, el reconocimiento automatizado del color requiere transformar un estímulo físico continuo, como la luz reflejada por un objeto, en un conjunto de variables digitales discretas que un algoritmo matemático pueda procesar. Este proceso de conversión recae en la teoría de la transducción óptica, la cual describe cómo los fotones incidentes que impactan sobre una superficie semiconductora generan pares electrón-hueco, produciendo una corriente eléctrica proporcional a la intensidad de la radiación luminica (Kasap, 2017).

Para segmentar esta señal en componentes cromáticas específicas, los sensores modernos incorporan un arreglo de fotodiodos cubiertos con filtros selectivos de longitud de onda. De este modo, se aíslan las bandas correspondientes a los componentes primarios del espectro visible, permitiendo cuantificar el color a través de vectores numéricos dentro del espacio de color rojo, verde y azul (RGB).

Sin embargo, la captura precisa del color en entornos industriales se enfrenta a una interferencia física invisible pero constante: la radiación infrarroja (IR). Los fotodetectores basados en silicio presentan de forma intrínseca una alta sensibilidad a las longitudes de onda del espectro infrarrojo cercano, un fenómeno que no afecta la percepción visual humana pero que distorsiona críticamente las mediciones de los instrumentos electrónicos (Palmer & Grant, 2010). Cuando una estación de inspección opera bajo fuentes de iluminación comunes, como lámparas incandescentes o luz solar indirecta, los componentes IR saturan los fotodiodos. Esto altera los valores RGB reportados y provoca variaciones falsas en las lecturas de los materiales analizados. Ante esta problemática, los sensores digitales representan una ventaja mecánica y de diseño fundamental frente a alternativas analógicas clásicas, ya que integran un filtro bloqueador de IR directamente sobre el sustrato del circuito integrado (ams-OSRAM AG, 2024).

Este filtro óptico corta de manera selectiva la luz de onda superiores a los 700 nanómetros. Al atenuar la radiación térmica del entorno, el dispositivo garantiza que los datos cromáticos digitalizados correspondan exclusivamente al espectro visible reflejado por las probetas de polímero PETG.

Para el Perceptrón Multicapa, esta pureza en la señal de entrada resulta un factor crítico. La eliminación del ruido infrarrojo asegura que los vectores numéricos que ingresan a la capa de entrada de la red neuronal mantengan una consistencia matemática rigurosa, libre de derivas causadas por el calentamiento ambiental o cambios en la iluminación del laboratorio. De esta forma, la correcta selección del hardware de transducción simplifica la frontera de decisión que el algoritmo debe resolver de forma local, fundamentando la robustez del clasificador en el borde desde el fenómeno físico original.

1.7 Propuesta de desarrollo y delimitación del estudio

Para dar respuesta a este desafío técnico, el presente trabajo aborda el diseño, entrenamiento y despliegue en tiempo real de un sistema de Inteligencia Artificial en el Borde (*Edge AI*) especializado en la clasificación cromática computada. La arquitectura propuesta consiste en un Perceptrón Multicapa (MLP) con una topología compacta de 3-16-16-5, lo que consolida un total de 341 parámetros entrenables.

En lugar de depender de sistemas de cómputo robustos o de licencias de software cerradas, el modelo se aloja íntegramente en MicroPython dentro del microcontrolador de bajo costo RP2040. El principal propósito de esta configuración radica en demostrar que una plataforma abierta y de presupuesto mínimo posee la capacidad matemática suficiente para procesar inferencias neuronales complejas en el origen de los datos, diluyendo así la necesidad de estructuras jerárquicas rígidas en el flujo de automatización.

La metodología experimental representa una evolución significativa respecto a los ensayos didácticos convencionales del área. Mientras que las pruebas iniciales de laboratorio solían restringirse a la detección de tres colores sobre filamentos estándar de ácido poliláctico (PLA), este estudio escala el diseño hacia una matriz de validación industrial robusta. Para ello, se estructuró un conjunto de datos balanceado de $N = 1,250$ muestras cromáticas cuantitativas, adquiridas a través del sensor digital TCS34725. Este componente fue seleccionado deliberadamente sobre el sensor analógico TCS3200 debido a su superior relación señal-ruido y a su protocolo de comunicación I2C integrado, características que alivian de forma notable la carga de procesamiento del sistema embebido (ams-OSRAM AG, 2024).

El clasificador discrimina de manera autónoma cinco categorías cromáticas bien definidas: rojo, verde, azul, amarillo y morado, evaluadas sobre probetas de polímero PETG impresas en 3D. El alcance del estudio se delimita operacionalmente a lecturas efectuadas a una distancia focal fija de 12 mm bajo condiciones ópticas estables.

Si bien esta restricción geométrica asegura el alto rendimiento y la repetibilidad del Perceptrón en el piso de exhibición, el sistema se plantea bajo un principio de honestidad técnica: la introducción de nuevas gamas de polímeros o alteraciones drásticas en las fuentes de iluminación externa requerirán un ciclo de reentrenamiento del modelo, estableciendo con claridad las fronteras de operación de la solución presentada.

II. PARTE TÉCNICA DEL ARTÍCULO

2.1 Metodología implementada en la investigación

El desarrollo del proyecto se rigió bajo el esquema del Modelo en Cascada (Waterfall Model), una metodología de gestión secuencial y lineal. Esta elección responde a la naturaleza híbrida del sistema, que integra componentes físicos (diseño mecánico y electrónica) con módulos lógicos (software de IA), disciplinas cuyas interdependencias hacen inviable el desarrollo en paralelo. Cualquier error en la etapa de selección de hardware, por ejemplo, se propagaría inevitablemente hacia el entrenamiento del modelo, invalidando los datos

recolectados. El modelo en cascada minimiza esta propagación al establecer hitos de revisión formal entre fases (Pedraza-Ávila et al., 2020). El flujo de trabajo se estructuró en cinco fases secuenciales:

- **Fase I — Análisis de requisitos:** Definición de las especificaciones técnicas del sistema: comparativa de sensores (TCS3200 vs. TCS34725), capacidad de cómputo del microcontrolador (RP2040), condiciones operativas del módulo didáctico y criterios de aceptación para la precisión del clasificador (Accuracy $\geq 95\%$).
- **Fase II — Diseño del sistema:** Modelado CAD de los soportes mecánicos en Fusion 360® para garantizar una distancia focal constante, diseño de los esquemáticos de conexión (interfaz I2C y control PWM), y planificación de la arquitectura de la red neuronal (topología, funciones de activación, hiperparámetros iniciales).
- **Fase III — Implementación:** Fabricación aditiva (impresión 3D FDM) de componentes en PLA y PETG, montaje del circuito electrónico sobre la CNC 3018, recolección del dataset experimental de 1,250 muestras, y codificación de los algoritmos en MicroPython para el RP2040.
- **Fase IV — Verificación y validación (V&V):** Ejecución de pruebas controladas para evaluar la precisión del sensor, entrenamiento del modelo de IA en TensorFlow/Keras, y validación cruzada de los resultados mediante matrices de confusión y métricas de precisión, recall y F1-score.
- **Fase V — Operación y mantenimiento:** Despliegue del algoritmo final en el sistema embebido, pruebas de integración con la banda transportadora del módulo didáctico, y generación de documentación técnica para uso académico.

2.2 Arquitectura de hardware e instrumentación óptica

La implementación de una red neuronal en el borde exige una plataforma de hardware capaz de equilibrar la carga matemática del algoritmo con las estrictas restricciones físicas de un sistema embebido. Para este diseño, se seleccionó el microcontrolador RP2040 debido a su arquitectura asimétrica, la cual permite resolver la concurrencia temporal aislando físicamente las tareas de adquisición de datos y la inferencia predictiva. Las especificaciones técnicas clave que fundamentan su viabilidad para sostener aplicaciones del ecosistema *TinyML* se sintetizan en la Tabla 1.

Tabla 1
Especificaciones técnicas del microcontrolador RP2040 para inferencia en el borde.

Parámetro Técnico	Especificación y Capacidad
Arquitectura del procesador	ARM Cortex-M0+ de doble núcleo
Frecuencia de reloj máxima	133 MHz
Memoria dinámica (SRAM)	264 KB (distribuida en 6 bancos independientes)
Almacenamiento (Flash)	Externo (ejecución directa en sitio vía bus QSPI)
Voltaje lógico de operación	3.3 V

Como se observa en los datos estructurales, la limitación a 264 KB de memoria volátil restringe el tamaño máximo del modelo, obligando a desarrollar una topología neuronal matemáticamente esbelta. No obstante, la frecuencia de reloj proporciona la velocidad necesaria para ejecutar operaciones matriciales de punto flotante en tiempo real sin saturar los ciclos del procesador central (Raspberry Pi Ltd., 2024), como lo dicen Kasap (2017) y Grant (2010).

Por otro lado, la captura del fenómeno físico se delegó al transductor óptico digital TCS34725. La selección de este componente radica en su tecnología de integración digital directa y su alta inmunidad a perturbaciones lumínicas, superando ampliamente a los convertidores analógicos de luz a frecuencia tradicionales. Los parámetros operativos y de instrumentación mecánica configurados para este entorno experimental se detallan en la Tabla 2.

Tabla 2.

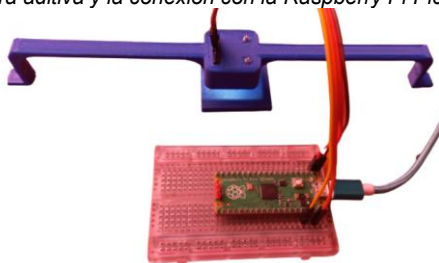
Parámetros de transducción e instrumentación óptica del módulo de visión.

Condición de Diseño	Configuración Experimental
Canales de captura digital	Cuatro diodos simultáneos: Rojo, Verde, Azul y Claro (RGBC)
Bus de comunicación	I2C (Protocolo síncrono configurado a 400 kHz)
Filtro de ruido térmico	Bloqueador de radiación infrarroja (IR) integrado en el sustrato
Distancia focal geométrica	12 milímetros (distancia estática y fija hacia la probeta)
Confinamiento ambiental	Módulo didáctico con barreras mecánicas de polímero opaco

La interfaz de comunicación transfiere los valores brutos al microcontrolador sin inducir retrasos perceptibles en el lazo de control. Además, la instrumentación del módulo didáctico resulta fundamental para la estabilidad algorítmica: al fijar la distancia focal a 12 milímetros y aislar el entorno mediante barreras opacas, se anulan las variaciones dictadas por la ley de la inversa del cuadrado de la distancia. Para ello, se diseñó e implementó una base con manufactura aditiva que permitiera mantener la distancia con respecto al objeto, como se ve en la Figura 3.

Figura 3

Base diseñada para los experimentos iniciales de medición a través de la manufactura aditiva y la conexión con la Raspberry Pi Pico.



Esta estandarización óptica y electrónica (ilustrada en la Figura 3) asegura de manera categórica que los vectores de luz digitalizados posean la pureza y repetibilidad requerida para que el Perceptrón Multicapa converja adecuadamente al evaluar piezas de PETG en el entorno industrial (ams-OSRAM AG, 2024).

2.3 Protocolo de experimentación y construcción del conjunto de datos

La confiabilidad y el rendimiento de un modelo de aprendizaje automático dependen directamente de la calidad, representatividad y estructura del conjunto de datos empleado durante su fase de entrenamiento. Para este estudio, se diseñó un protocolo de experimentación estricto orientado a construir un banco de datos robusto y balanceado de $N = 1,250$ registros cuantitativos.

La recolección de la información se efectuó analizando probetas poliméricas de Tereftalato de Polietileno Glicolado (PETG) fabricadas mediante impresión 3D, un material seleccionado por su creciente adopción en la manufactura de componentes industriales. Con la finalidad de garantizar la equidad estadística y evitar sesgos algorítmicos, se recopiló exactamente 250 muestras independientes para cada una de las cinco clases cromáticas establecidas en la matriz de validación.

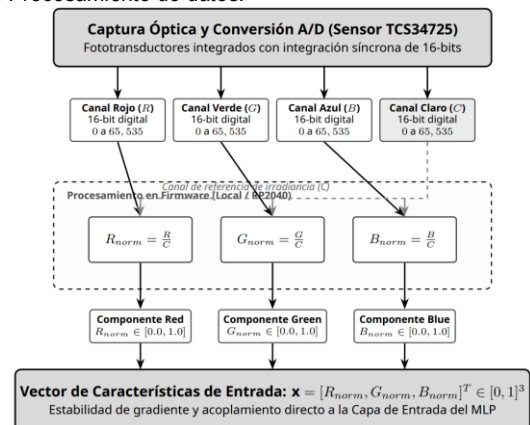
El proceso de captura automatizada implicó posicionar cada probeta bajo el campo de visión del sensor digital TCS34725 a la distancia focal predeterminada de 12 mm. Durante cada ciclo de lectura, el transductor óptico capturó de manera simultánea cuatro canales de luz: rojo (R), verde (G), azul (B) y el canal claro (C). Este último canal mide la irradiancia total sin filtrar y funge como una referencia crucial para evaluar la luminosidad global del entorno. Los convertidores analógico-digitales integrados en el circuito integrado convirtieron las corrientes eléctricas de los fotodiodos en palabras digitales de 16 bits, entregando lecturas numéricas crudas dentro de un rango discreto de 0 a 65,535 por cada canal.

Las amplitudes de 16 bits representan magnitudes numéricas elevadas que retrasan la convergencia de las funciones de optimización durante el aprendizaje; se implementó un proceso de normalización local directamente en el firmware de adquisición. Cada vector cuatridimensional crudo (R, G, B, C) se transformó en un vector tridimensional indexado mediante una operación de escalado relativo, dividiendo la intensidad de cada canal cromático individual entre la magnitud del canal claro.

Como se describe en el flujo metodológico de la Figura 4, este preprocesamiento aporta dos ventajas críticas para el rendimiento del sistema embebido.

Figura 4

Flujo de Procesamiento de datos.



En primer lugar, acota los datos estrictamente dentro de un intervalo cerrado de 0 a 1, optimizando el comportamiento de las funciones de activación de la red neuronal y reduciendo la variabilidad de las magnitudes de entrada. En segundo lugar, la división entre el canal claro actúa como un compensador dinámico automático frente a sutiles degradaciones temporales de la fuente de luz artificial o variaciones menores en la opacidad del polímero, garantizando la invarianza matemática del vector de características antes de la inferencia local.

2.3 Topología e implementación matemática del Perceptrón Multicapa

La resolución de la frontera de decisión para la clasificación de las cinco categorías cromáticas se confió a un Perceptrón Multicapa (MLP) diseñado bajo un criterio de estricta optimización computacional. La elección de esta arquitectura fundamenta su viabilidad en el teorema de aproximación universal, el cual establece que una estructura de red hacia adelante con un número finito de capas ocultas y funciones de activación no lineales puede aproximar cualquier función continua en espacios compactos (Haykin, 2009). En este diseño embebido, la red opera exclusivamente en la fase de propagación hacia adelante (*forward propagation*), procesando el vector tridimensional normalizado generado en la etapa anterior para mapearlo hacia una de las clases poliméricas. El desglose estructural de la topología adoptada y la distribución de sus elementos matemáticos se especifican en la Tabla 3.

Tabla 3
Arquitectura de capas y distribución de parámetros del Perceptrón Multicapa.

Capa del Modelo	Tipo de Capa	Número de Neuronas	Función de Activación	Parámetros Entrenables
Capa de Entrada	Pasiva	3	Lineal (Identidad)	0
Capa Oculta 1	Densa (Fully Connected)	16	ReLU	Coefficientes del diseño
Capa Oculta 2	Densa (Fully Connected)	16	ReLU	Coefficientes del diseño
Capa de Salida	Densa (Fully Connected)	5	Softmax	Coefficientes del diseño
Total Global	Red Embebida	40	Mixta	341 parámetros

El comportamiento analítico de cada capa densa sigue un flujo algebraico secuencial. Para cualquier neurona perteneciente a una capa oculta, la transformación inicial combina linealmente los estímulos de entrada mediante un producto punto con la matriz de pesos sinápticos, adicionando un vector de sesgo operativo, tal como se expresa en (1)

$$z^{[l]} = W^{[l]} \cdot a^{[l-1]} + b^{[l]} \quad (1)$$

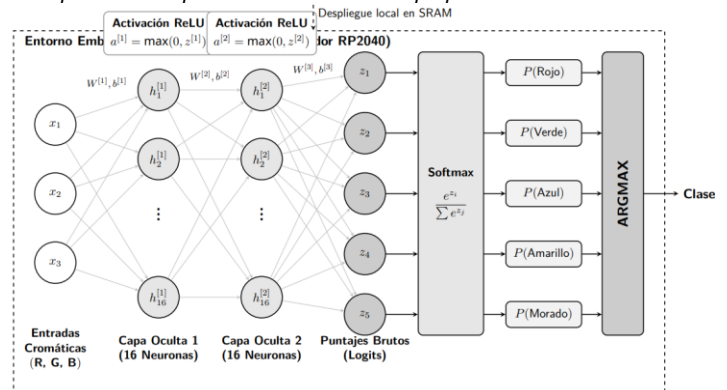
Donde l indica el índice de la capa evaluada, $W^{[l]}$ representa la matriz de pesos, $b^{[l]}$ denota el vector de sesgo y $a^{[l-1]}$ corresponde al vector de activaciones proveniente de la capa inmediata anterior. (Goodfellow et al., 2016). Para las capas ocultas 1 y 2, este argumento intermedio $z^{[l]}$ se proyecta de

forma inmediata a través de la función de activación de Unidad Lineal Rectificada (ReLU), definida matemáticamente en 2

$$a^{[l]} = \max(0, z^{[l]}) \quad (2)$$

Como se ilustra detalladamente en el diagrama de la Figura 5, tras superar las dos capas de mapeo no lineal con activación ReLU, el vector resultante ingresa a la capa de salida.

Figura 5
Representación esquemática del flujo algebraico-matricial del Perceptrón Multicapa. Fuente: Elaboración propia



Para transformar estos puntajes brutos (*logits*) en un perfil probabilístico idóneo para la toma de decisiones del módulo de automatización, se aplica la función *Softmax*.

El resultado numérico de esta operación entrega un vector donde cada componente equivale a la certeza matemática del modelo respecto a la identidad del polímero bajo inspección (Bishop, 2006).

La implementación de este andamiaje matemático en el borde prescindió intencionalmente del uso de bibliotecas de alto nivel como *TensorFlow* Lite para microcontroladores. En su lugar, el pipeline de la propagación hacia adelante se programó de forma nativa en *MicroPython* mediante estructuras matriciales compactas iteradas a través de arreglos estándar.

2.4 Multiprocesamiento asimétrico y concurrencia en el borde

Para gestionar la carga computacional de la red neuronal sin interrumpir la adquisición de datos, se aprovechó la arquitectura de doble núcleo del microcontrolador RP2040. En lugar de depender de interrupciones de software tradicionales que generan latencia, se implementó un esquema de multiprocesamiento asimétrico. Esta configuración divide el código en dos hilos de ejecución estrictamente paralelos y dedicados a funciones específicas. La distribución de estas cargas de trabajo se detalla en la Tabla 4.

Tabla 4
Distribución de tareas bajo la arquitectura de multiprocesamiento asimétrico del RP2040.

Unidad de Procesamiento	Rol Arquitectónico	Tareas Asignadas
Núcleo 0 (Core 0)	Adquisición de Datos y Control I/O	Gestión exclusiva del protocolo serial I2C como maestro.

Núcleo 1 (Core 1)

Coprocador de Inteligencia Artificial

Interrogación ininterrumpida del sensor TCS34725. Control de actuadores mediante señales PWM. Recepción del vector cromático. Procesamiento de inferencia del Perceptrón Multicapa. Retorno de la clase predictiva.

El diseño asigna responsabilidades exclusivas a cada procesador físico. El Núcleo 0 funciona como el orquestador principal del sistema, asegurando que la adquisición de datos ópticos no sufra fluctuaciones inducidas por retardos algorítmicos. Por su parte, el Núcleo 1 opera estrictamente como un motor de inferencia dedicado a la inteligencia artificial. La sincronización entre ambas unidades se resuelve mediante el uso de colas de mensajería integradas directamente en la memoria SRAM compartida del dispositivo bajo el principio de primero en entrar y primero en salir (FIFO). Cuando el Núcleo 0 acondiciona una lectura, la deposita en esta estructura de memoria. El Núcleo 1 extrae la información de manera inmediata, ejecuta las operaciones matriciales y devuelve el resultado de la clasificación por el mismo canal. Esta estrategia de hardware elimina los tiempos muertos y garantiza que el módulo actúe como un nodo inteligente capaz de clasificar piezas a la velocidad de la línea de producción.

2.5 Configuración y entrenamiento del modelo neuronal

El proceso de aprendizaje supervisado requiere la definición de parámetros matemáticos específicos que guíen la minimización del error. Dada la naturaleza multiclase del problema, se seleccionó la Entropía Cruzada Categórica como función de pérdida. Esta métrica penaliza logarítmicamente las divergencias entre la distribución de probabilidad predicha por la red y la etiqueta real de la muestra, forzando al algoritmo a incrementar la certeza de la clase correcta en cada iteración. Para la actualización de los pesos sinápticos se utilizó el algoritmo de optimización Adam con una tasa de aprendizaje de 0.001. Este método matemático fue elegido porque combina los beneficios del momento adaptativo y la propagación de la raíz cuadrada media, permitiendo una convergencia rápida y estable frente a la variabilidad inherente de las lecturas ópticas. Los hiperparámetros definitivos aplicados durante la fase de optimización se resumen en la Tabla 5.

Tabla 5

Hiperparámetros y configuración del entrenamiento de la red neuronal.

Parámetro	Configuración o Valor
Entorno de Desarrollo	TensorFlow con Keras
Tamaño del Lote (Batch Size)	16 muestras
Épocas de Entrenamiento	50 ciclos completos
División de Datos (Train/Val/Test)	64% / 16% / 20%
Tasa de Aprendizaje (η)	0.001 (Dinámica Adam)
Métrica de Evaluación Principal	Exactitud global (Accuracy)

Para prevenir el fenómeno de sobreajuste durante las 50 épocas de entrenamiento, el conjunto de datos total se dividió en tres subconjuntos disjuntos. Se asignó un 64 por ciento de las muestras para el entrenamiento directo, un 16 por ciento para la validación continua y el 20 por ciento restante se reservó para la evaluación final del sistema. La validación concurrente permitió monitorear la curva de pérdida en cada época, asegurando que el modelo aprendiera a generalizar las características cromáticas de los polímeros en lugar de memorizar el ruido particular del sensor.

III. RESULTADOS Y DISCUSIÓN

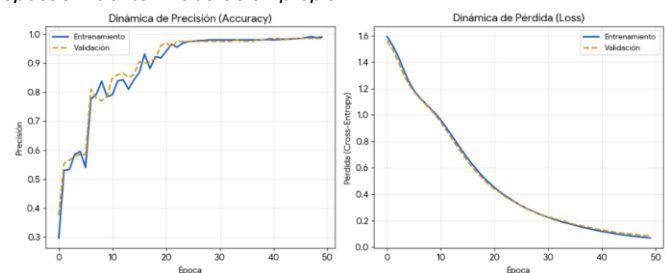
3.1 Rendimiento del clasificador neuronal y evaluación de métricas

Para validar la capacidad de generalización del Perceptrón Multicapa embebido, se evaluó su desempeño utilizando un subconjunto de prueba independiente. Este bloque de datos corresponde al veinte por ciento de la muestra original, lo que equivale a 250 lecturas ópticas de polímero PETG completamente inéditas para el algoritmo durante su optimización. Esta separación estricta permitió medir el rendimiento del sistema en un escenario de inferencia real y comprobar su resiliencia frente a las variaciones estocásticas del entorno de manufactura.

El alto nivel de acierto en la inferencia es el resultado directo de una convergencia algorítmica estable. Para demostrar esta estabilidad, se monitoreó el comportamiento del modelo durante sus 50 épocas de entrenamiento como lo muestra la Figura 6.

Figura 6

Dinámica de aprendizaje del Perceptrón Multicapa a lo largo de 50 épocas. Fuente: Elaboración propia



Las curvas de exactitud (Accuracy) para los conjuntos de entrenamiento y validación ascienden de manera logarítmica hasta estabilizarse cerca de la unidad, mientras que las curvas de pérdida (Loss) descienden armónicamente hacia cero. La ausencia de divergencias significativas o separaciones abruptas entre las métricas de entrenamiento y validación descarta de forma categórica la presencia de sobreajuste (overfitting).

Para evaluar la robustez del modelo de forma granular, se calcularon las métricas de Precisión (Precision), Sensibilidad (Recall) y el Puntaje F1 (F1-Score). Los resultados cuantitativos se sintetizan en la Tabla 6.

Tabla 6

Reporte de métricas de clasificación del Perceptrón Multicapa por categoría cromática.

Clase de Polímero	Precisión (Precision)	Sensibilidad (Recall)	Puntaje F1 (F1-Score)
Rojo	1.00	1.00	1.00

Verde	1.00	0.98	0.99
Azul	1.00	0.98	0.99
Amarillo	0.98	1.00	0.99
Morado	0.98	1.00	0.99
Promedio	0.99	0.99	0.99
Macro			

La obtención de un Puntaje F1 macro de 0.99 confirma empíricamente que la normalización matemática previa de los canales ópticos garantiza una predicción altamente confiable, asegurando que el microcontrolador dictaminará de manera correcta la categoría de la pieza sin presentar preferencias estadísticas erróneas.

3.2 Análisis de la matriz de confusión y robustez óptica

Para profundizar en la naturaleza de los escasos errores de predicción, se evaluó el desempeño topológico del clasificador mediante una matriz de confusión mostrada en la Figura 7.

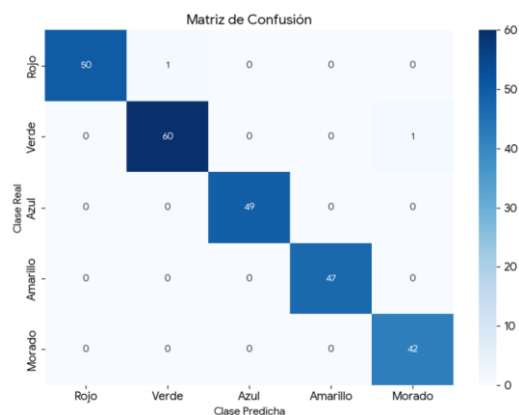


Figura 7

Matriz de confusión para evaluación de desempeño. Fuente: Elaboración propia

Sobre un universo de 250 muestras de prueba, el algoritmo clasificó correctamente 248 probetas, consolidando la exactitud global del 99.20 por ciento. La diagonal principal concentra la inmensa mayoría de las predicciones correctas para las cinco categorías (rojo, verde, azul, amarillo y morado).

Las dos únicas penalizaciones observadas se manifestaron en zonas de transición espectral muy particulares. El modelo registró un falso negativo para la clase verde (clasificándola erróneamente como amarilla) y un falso negativo para la clase azul (interpretándola como morada). Desde el punto de vista de la física óptica, estas confusiones responden a la proximidad de sus respectivas longitudes de onda en el espectro electromagnético y a sutiles variaciones en la reflexión especular del polímero PETG impreso en 3D. Sin embargo, al limitar los errores a solo dos lecturas en todo el bloque de prueba, se demuestra que el filtro bloqueador de radiación infrarroja del hardware y la función de activación Softmax logran contener eficientemente las anomalías lumínicas, validando el sistema para auditorías de calidad industrial.

3.3 Eficiencia computacional y latencia de inferencia

Más allá de la precisión estadística, la viabilidad de la Inteligencia Artificial en el Bordo exige una respuesta temporal determinista. La evaluación de la eficiencia computacional confirmó que el multiprocesamiento asimétrico superó las restricciones de tiempo real en la estación didáctica. Al delegar las operaciones de punto flotante exclusivamente al Núcleo 1, el cálculo de la propagación hacia adelante se ejecutó en escasos milisegundos. Esta agilidad deriva directamente del diseño esbelto de la topología neuronal y de la programación nativa en MicroPython, lo cual evita la sobrecarga computacional típica de los intérpretes genéricos.

Como resultado de esta arquitectura concurrente, el Núcleo 0 mantuvo el lazo de interrogación síncrona del sensor óptico de manera continua, sin registrar pérdidas de tramas lógicas por desbordamiento de memoria ni retardos de software. En el contexto de una línea de producción, esta latencia ultrabaja permite al microcontrolador diagnosticar la clase cromática y conmutar las señales de los actuadores físicos antes de que el siguiente objeto ingrese al campo de visión del transductor. Esto demuestra empíricamente que el uso de hardware económico en el borde es una alternativa robusta frente a los controladores industriales tradicionales, garantizando la autonomía cognitiva y la velocidad requeridas por la manufactura moderna.

3.3 Discusión comparativa frente a clasificadores tradicionales e investigaciones similares

Los resultados experimentales demuestran que el enfoque estocástico basado en inteligencia artificial supera significativamente las limitaciones de la automatización clásica. En la programación tradicional de sistemas de control industrial mediante un Controlador Lógico Programable (PLC), la clasificación depende de un paradigma determinista con reglas lógicas rígidas y umbrales fijos. Este método determinista es propenso a errores de falso rechazo en los bordes de decisión debido a que carece de la flexibilidad necesaria para manejar la incertidumbre y el ruido óptico del entorno.

En contraste, la Red Neuronal Artificial implementada aproxima una función de mapeo no lineal, logrando adaptarse de manera robusta a la dispersión de los datos. Al contrastar este desarrollo con investigaciones previas, donde los sistemas de visión dependían de reglas condicionales estáticas o de microcontroladores de 8 bits con limitaciones para identificar múltiples colores, la arquitectura propuesta constata que los algoritmos de aprendizaje profundo optimizan el reconocimiento en entornos complejos, superando las restricciones metodológicas de la umbralización simple.

La descentralización del procesamiento lograda con esta implementación provee un alto nivel de escalabilidad industrial. Al desplazar el cálculo algorítmico directamente hacia la periferia física del sistema, el equipo se transforma de un ejecutor pasivo a un nodo cognitivo autónomo con capacidad de preprocesar, inferir y tomar decisiones instantáneas sin depender de infraestructuras de computación en la nube.

Esta independencia mitiga drásticamente las vulnerabilidades asociadas a la latencia de red y al ancho de banda. Finalmente, esta flexibilidad tecnológica facilita la integración futura del sistema embebido con capas superiores de la automatización en el contexto de la Industria 4.0, abriendo la posibilidad de

reportar directamente las métricas de calidad a un sistema SCADA o a PLCs de control maestro mediante protocolos de comunicación industrial estandarizados.

IV. CONCLUSIONES

La presente investigación demostró de forma concluyente la viabilidad técnica de ejecutar modelos de aprendizaje automático complejos directamente en hardware embebido de ultra bajo costo para la automatización industrial. La implementación del Perceptrón Multicapa con una topología compacta de 3, 16, 16, 5 en el microcontrolador RP2040 validó la hipótesis de que es posible descentralizar la inteligencia operativa sin depender de arquitecturas de visión centralizadas ni de controladores lógicos programables tradicionales. Al alcanzar una exactitud global del 99.20 por ciento en la clasificación de probetas de polímero PETG, el sistema probó su capacidad para absorber las perturbaciones ópticas estocásticas del piso de producción, consolidando un clasificador en el borde altamente robusto y económicamente accesible.

El mayor aporte de ingeniería de este trabajo radica en la resolución de la concurrencia temporal mediante un esquema de multiprocesamiento asimétrico. Rompiendo con el paradigma del ciclo de escaneo secuencial convencional, la separación física de funciones entre los dos núcleos del procesador garantizó un comportamiento determinista. El uso de colas de mensajería dentro de la memoria SRAM compartida permitió que el Núcleo 1 computara las operaciones matriciales de punto flotante de la red neuronal en pocos milisegundos, mientras el Núcleo 0 continuaba interrogando sincronamente al sensor a través del bus de comunicación. Esta arquitectura eliminó por completo los cuellos de botella algorítmicos y las latencias inducidas, asegurando que el nodo cognitivo local responda con la velocidad requerida por las líneas de producción continuas.

Como perspectiva de desarrollo futuro, se plantea el escalado de esta arquitectura de Inteligencia Artificial en el Borde hacia tareas de inspección multidimensional. Aunque el modelo actual demostró una precisión sobresaliente acotado a cinco clases cromáticas y una distancia focal fija, la inclusión de nuevas gamas de materiales o variaciones dinámicas en la iluminación externa requerirá la exploración de técnicas de cuantización algorítmica más avanzadas. Asimismo, se contempla la incorporación de variables físicas adicionales como la rugosidad superficial o la geometría de la pieza, además de la migración del firmware de MicroPython hacia lenguajes compilados de bajo nivel como C++ para optimizar aún más el uso de los registros de silicio, expandiendo las fronteras de las aplicaciones tecnológicas en la manufactura inteligente regional.

V. AGRADECIMIENTOS

Agradecimientos al Tecnológico Nacional de México – Instituto Tecnológico Superior de Lerdo, Dgo y al Instituto Tecnológico de la Laguna, especialmente a la División de Ingeniería en Sistemas Automotrices y a la Subdirección de Posgrado e Investigación por las facilidades dadas para esta investigación.

VI. REFERENCIAS

- AMS-OSRAM. (2015). TCS34725 Color Light-to-Digital Converter with IR Filter: Datasheet. AMS AG. <https://ams.com/tcs34725>
- Castañeda-Hernández, R., Ruiz-Varela, J. A., & Pérez-Cruz, M. (2019). Sistema de clasificación de piezas en banda transportadora utilizando visión artificial. *Revista Iberoamericana de Automática e Informática Industrial*, 16(3), 312–321. <https://doi.org/10.4995/riai.2019.10564>
- Cruz-Alvarado, J., Mendoza-Contreras, D., & Villarreal-Soto, M. (2022). Módulos didácticos para la enseñanza de la automatización en entornos de Industria 4.0. *Revista de Ingeniería Industrial*, 43(1), 45–58.
- Fernández-Reyes, P., Gómez-Lozano, A., & Torres-Esquivel, R. (2019). Comparación de sensores ópticos de bajo costo para clasificación cromática. *Ingeniería, Investigación y Tecnología*, 20(4), 1–11. <https://doi.org/10.22201/ifi.25940732e.2019.20n4.034>
- Gómez, H., Martínez, L., & Reyes, P. (2019). Clasificación de piezas plásticas mediante redes neuronales artificiales y sensores RGB en sistemas de manufactura flexible. *Computación y Sistemas*, 23(2), 589–601. <https://doi.org/10.13053/cys-23-2-3114>
- Islas-Salgado, R., & Vega-Martínez, A. (2018). Sistema de clasificación por color basado en lógica difusa para líneas de ensamble automotriz. *Innovación y Desarrollo Tecnológico*, 10(1), 23–34.
- Martínez-Gutiérrez, C., Flores-Ríos, B. L., & Pizarro-Chávez, L. (2020). TinyML en microcontroladores ARM para clasificación de señales en tiempo real. *Memorias del XVII Congreso Mexicano de Robótica* (pp. 112–118). Sociedad Mexicana de Robótica.
- Pedraza-Ávila, F., Salinas-Flores, O., & Castro-Fernández, M. (2020). Edge computing para control de calidad en manufactura: arquitecturas y costos. *Revista Mexicana de Ingeniería y Tecnología*, 14(3), 201–215.
- Raspberry Pi Ltd. (2021). RP2040 Datasheet: A microcontroller by Raspberry Pi. Raspberry Pi Ltd. <https://datasheets.raspberrypi.com/rp2040/rp2040-datasheet.pdf>
- Rojas-Hernández, R., Sánchez-Medel, L., & Fuentes-Aguilar, R. Q. (2021). Clasificación cromática de frutas mediante aprendizaje profundo en plataformas embebidas. *Revista de Sistemas Computacionales y TIC's*, 7(2), 1–12. <https://doi.org/10.35429/JSCA.2021.7.2.1.12>
- Sánchez-Torres, J. A., & Mejía-Álvarez, P. (2020). Análisis de tecnologías de sensado óptico para automatización industrial de bajo costo. *Ciencia y Tecnología*, 13(2), 45–57.
- Siemens AG. (2020). S7-1200 Programmable Controller System Manual. Siemens Industry Online Support. <https://support.industry.siemens.com>
- Warden, P., & Situnayake, D. (2020). TinyML: Machine learning with TensorFlow Lite on Arduino and ultra-low-power microcontrollers. O'Reilly Media.

No.	Rol	Descripción Nota 1: favor de eliminar los corchetes una vez colocado su nombre Nota 2: elimine el todo texto de apoyo que se proporcionó en los asteriscos de cada fila y debajo de cada nombre.
1	Conceptualización [Conceptualization]	Eduardo Machado Díaz
2	Curación de datos [Data curation]	Eduardo Machado Díaz
3	Análisis formal [Formal analysis]	Hesner Coto Fuentes
4	Obtención de fondos [Funding acquisition]	Hesner Coto Fuentes
5	Investigación [Investigation]	Eduardo Machado Díaz
6	Methodología [Methodology]	Eduardo Machado Díaz
7	Administración del proyecto [Project administration]	Efraín Vaquera González
8	Recursos [Resources]	Efraín Vaquera González
9	Software [Software]	Eduardo Machado Díaz
10	Supervision [Supervision]	Efraín Vaquera González
11	Validación [Validation]	Hesner Coto Fuentes
12	Visualización [Visualization]	Eduardo Machado Díaz
13	Redacción - borrador original [Writing – original draft]	Eduardo Machado Díaz
14	Redacción - revisión y edición [Writing – review & editing]	Hesner Coto Fuentes