



UNIVERSIDAD DE GUANAJUATO

---

---

CAMPUS IRAPUATO-SALAMANCA  
DIVISIÓN DE INGENIERÍAS

*“Herramienta de apoyo en diagnóstico  
preventivo de pie plano utilizando  
procesamiento de imágenes de profundidad.”*

## TESIS PROFESIONAL

QUE PARA OBTENER EL GRADO DE:  
*MAESTRO EN INGENIERÍA ELÉCTRICA*

PRESENTA:

*Ing. Jessica Nohemí Figueroa Ramírez*

DIRECTORES DE TESIS:

*Dr. Ángel Díaz Pacheco*  
*Dr. Juan Manuel Sierra Hernández*

SALAMANCA, GTO.

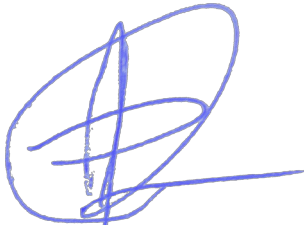
Mayo, 2026

Salamanca, Gto., a 24 de Marzo del 2026.

**MTRO. JUAN SIGFRIDO LÓPEZ CUSTODIO**  
**COORDINADOR DE ASUNTOS ESCOLARES**  
**P R E S E N T E.-**

Por medio de la presente, se otorga autorización para proceder a los trámites de impresión, empastado de tesis y titulación al alumno(a) Jessica Nohemí Figueroa Ramírez del **Programa de Maestría en Ingeniería Eléctrica** y cuyo número de **NUA** es: 206239 del cual soy director. El título de la tesis es: Herramienta de apoyo en diagnóstico preventivo de pie plano utilizando procesamiento de imágenes de profundidad.

Hago constar que he revisado dicho trabajo y he tenido comunicación con los sinodales asignados para la revisión de la tesis, por lo que no hay impedimento alguno para fijar la fecha de examen de titulación.



DR. ÁNGEL DÍAZ PACHECO  
**DIRECTOR DE TESIS**  
**SECRETARIO**


**A T E N T A M E N T E**



DR. JUAN MANUEL SIERRA HERNÁNDEZ  
**DIRECTOR DE TESIS**



DR. ÓSCAR GERARDO IBARRA MANZANO  
**PRESIDENTE**



DR. JOSÉ RUIZ PINALES  
**VOCAL**

---

## Agradecimientos Institucionales

A la Universidad de Guanajuato por el apoyo recibido para el desarrollo del proyecto.



Al SECIHTI a través del Programa de Becas Nacionales por la beca de estudiante núm. 2002199.



**Ciencia y Tecnología**  
Secretaría de Ciencia, Humanidades, Tecnología e Innovación

---

# Agradecimientos

A mis padres, José Jesús Figueroa Montenegro y Alma Alejandra Ramírez Cerda. A mi papá, por su apoyo incondicional a lo largo de mi vida; y a mi mamá, por su hermosa forma de enseñarme a ver la vida aun en medio de las adversidades. Tenerlos a ambos ha sido un gran regalo y el mejor ejemplo para seguir adelante.

A mi asesor, el Dr. Ángel Díaz Pacheco, por sus valiosas retroalimentaciones para lograr un trabajo de calidad, así como por todos los aprendizajes obtenidos. Sus consejos y su forma de enseñar me permitieron adquirir herramientas y formas de análisis que fueron fundamentales para desarrollar este trabajo.

A mi coasesor, el Dr. Juan Manuel Sierra Hernández, perteneciente al Cuerpo Académico de Telecomunicaciones y Fotónica, por compartir conmigo la idea de implementar un sistema para el análisis del pie plano, así como por su apoyo económico para la adquisición del equipo y la estructura necesarios para su desarrollo. Agradezco también sus consejos académicos, como la recomendación de cursar materias clave durante la maestría.

De manera especial agradezco a la Dra. Dora Luz Almanza, por su excelente curso de visión por computadora, el cual aportó conocimientos fundamentales para el desarrollo de este proyecto. Asimismo, al Dr. Pinales, quien además de formar parte de los sinodales de este trabajo, impartió la materia de redes neuronales; gracias por su gran labor docente y por sus valiosas correcciones que permitieron fortalecer esta investigación.

Agradezco también al Dr. Óscar Gerardo, por los conocimientos que ha contribuido a formar en mí desde la licenciatura. De manera particular, reconozco su desempeño como director de departamento. Gracias a una beca obtenida mediante convocatoria, y al apoyo complementario brindado por la Universidad de Guanajuato, pude realizar un viaje académico a Singapur, experiencia que sin duda enriqueció profundamente mi formación académica y personal.

Al Dr. Julián Moisés Estudillo Ayala y al Dr. Roberto Rojas Laguna, pertenecientes al Cuerpo Académico de Telecomunicaciones y Fotónica, gracias por su apoyo desde la licenciatura hasta la maestría y por mantener en pie el capítulo estudiantil IEEE Photonics UG.

A mis amigos de tenis de mesa, porque más que un deporte fue una gran terapia para sobrellevar lo complejo que puede ser terminar un posgrado. Fueron un respiro y un abrazo para el alma en momentos que lo necesitaba.

A mis amigos Alejandro y Roberto. Aunque proveníamos de diferentes licenciaturas y teníamos distintos enfoques, justamente esa diversidad nos permitió cuestionar y analizar los problemas desde múltiples perspectivas. Sus ideas, comentarios y retroalimentaciones me ayudaron a ver los retos de muchas maneras posibles. Compartir este tiempo con ustedes fue una experiencia invaluable; tienen todo mi cariño.

A mi esposo, quien siempre ha estado y sigue estando a mi lado. Te amo, gracias por acompañarme en este camino, por ser un pilar en mi vida y por la determinación y disciplina que te caracterizan. Admiro profundamente el gran ingeniero y ser humano que eres, y agradezco a la vida por haber cruzado nuestros caminos. Estoy segura de que muy pronto también serás el mejor papá.

Y finalmente, a Olivia, que aun en medio de todos los retos que implicó concluir este grado académico, se aferró a mí con cada parte de su ser. Mi vida, espero que después de defender este trabajo sigas aquí conmigo. No te sueltes... ya casi estás en nuestros brazos.

## Abstract

*The medial longitudinal arch plays a key role in load distribution during gait, and its alteration has been associated with discomfort, chronic pain and other foot disorders. In clinical practice, flatfoot assessment usually relies on 2D footprint indices or on more expensive instrumented systems. This thesis proposes a methodology to analyse plantar footprints using depth maps and machine learning techniques, with the aim of exploring an affordable tool to support the early detection of flatfoot in adults.*

*A custom acquisition setup was built using an Intel RealSense camera and a glass platform, enabling the simultaneous capture of RGB images and depth maps of the footprints. The raw depth data were processed through a plane correction algorithm, footprint segmentation combining depth and colour information, and the computation of morphological descriptors such as projected area, volume under the arch and average height. In parallel, traditional clinical indices (Staheli and Chippaux-Smirak) were estimated, and normalised footprint images were generated as input to a convolutional neural network used to extract latent embeddings.*

*Several binary classification experiments (flatfoot vs. non-flatfoot) were conducted, comparing Bayes models with kernel density estimation (KDE), multilayer perceptrons and classical classifiers trained either on full depth matrices or on the extracted embeddings. The results indicate that working with compact latent representations, together with explicit class balancing using the synthetic minority over-sampling technique (SMOTE), substantially improves the sensitivity to the flatfoot class while preserving high overall performance. These findings suggest that combining depth sensing, morphological descriptors and machine learning models offers a promising and low-cost complementary approach for quantitative analysis of the plantar arch.*

## Resumen

*El arco plantar desempeña un papel importante en la distribución de cargas durante la marcha y su alteración puede estar relacionada con molestias, dolor crónico*

y otras patologías. En la práctica clínica, la detección del pie plano suele apoyarse en índices obtenidos a partir de huellas plantares 2D o en estudios instrumentados de mayor costo. En este trabajo se propuso una metodología para analizar huellas plantares mediante mapas de profundidad y técnicas de aprendizaje automático, con el fin de explorar una alternativa accesible para el apoyo al diagnóstico preventivo del pie plano en población adulta.

La base de datos se obtuvo con una cámara Intel RealSense y un sistema de captura diseñado para este propósito, que permite registrar de manera simultánea información RGB y de profundidad de las huellas sobre una superficie de vidrio. A partir de los mapas de profundidad, se aplicó una etapa de preprocesamiento que incluye corrección de plano, segmentación de la huella mediante información de color y profundidad, y cálculo de métricas morfológicas como área proyectada, volumen bajo el arco y altura promedio. De forma complementaria se calcularon índices clínicos tradicionales (Staheli y Chippaux-Smirak) y se construyeron imágenes normalizadas para alimentar un extractor de vectores de características basado en redes neuronales convolucionales.

Con estos insumos se llevaron a cabo diversos experimentos de clasificación binaria (pie plano / pie no plano), comparando modelos bayesianos basados en estimación de densidad por núcleo (kernel density estimation, KDE), perceptrones multicapa y clasificadores clásicos entrenados sobre matrices de profundidad completas y sobre embeddings. Los resultados muestran que el uso de representaciones latentes de dimensión moderada, combinado con estrategias explícitas de balanceo de clases como sobremuestreo de la clase minoritaria sintética (synthetic minority over-sampling technique, SMOTE), mejora de manera notable la sensibilidad hacia la clase pie plano sin sacrificar el desempeño global. Estos hallazgos sugieren que la integración de sensores de profundidad, descriptores morfológicos y modelos de aprendizaje automático puede convertirse en una herramienta complementaria de bajo costo para el estudio cuantitativo del arco plantar.

---

# Índice general

<b>1. Introducción</b>	<b>1</b>
1.1. Justificación . . . . .	2
1.2. Objetivos . . . . .	3
1.2.1. General . . . . .	3
1.2.2. Específicos . . . . .	3
1.3. Hipótesis . . . . .	4
1.4. Productos obtenidos . . . . .	4
1.5. Organización . . . . .	5
<b>2. Antecedentes y Estado del Arte</b>	<b>8</b>
2.1. Conceptos y definiciones clínicas del pie . . . . .	8
2.2. Métodos diagnósticos en la práctica clínica . . . . .	11
2.3. Métodos computacionales aplicados al diagnóstico del pie plano . . .	17
2.3.1. Clasificación basada en el análisis morfológico 2D . . . . .	17
2.3.2. Clasificación basada en datos de presión plantar . . . . .	20
2.3.3. Modelado 3D y Escaneo Tridimensional: La Base de los Enfoques de Vanguardia . . . . .	21
2.4. Limitaciones de los enfoques revisados . . . . .	23
2.5. Brecha encontrada . . . . .	23
<b>3. Marco teórico</b>	<b>25</b>
3.1. Introducción a la visión por computadora con profundidad . . . . .	26
3.2. Principios físicos de la obtención de imágenes de profundidad . . . .	27
3.2.1. Métodos comunes para obtener profundidad . . . . .	28
3.2.1.1. Luz estructurada vs luz codificada . . . . .	29
3.2.1.2. Profundidad estéreo . . . . .	30
3.2.1.3. Tiempo de vuelo . . . . .	30
3.3. Funcionamiento de la Cámara Intel RealSense D435i . . . . .	31
3.3.1. Arquitectura del sistema estéreo: . . . . .	31
3.3.2. Emisión infrarroja y seguridad ocular . . . . .	32

3.4.	Matemática detrás de la visión estéreo . . . . .	33
3.4.1.	Matriz de profundidad y triangulación estereoscópica . . . . .	33
3.5.	Transformación de Datos a nube de puntos . . . . .	35
3.5.1.	Formato de archivo: .ply . . . . .	35
3.6.	Componentes y parámetros de un sistema de visión . . . . .	38
3.6.1.	Profundidad de campo (depth of field, DoF) . . . . .	38
3.6.2.	Campo de visión (field of view, FOV) . . . . .	39
3.6.3.	Resolución . . . . .	40
3.6.4.	Distancia mínima y máxima de trabajo . . . . .	43
3.7.	Software Intel RealSense (Software Development Kit, SDK) . . . . .	43
3.8.	Segmentación basada en color . . . . .	44
3.8.1.	Fundamentos de los espacios de color . . . . .	45
3.8.2.	Espacio de color HSL y su uso . . . . .	46
3.9.	Redes neuronales y aprendizaje profundo con aplicación a visión por computadora . . . . .	47
3.9.1.	Fundamentos de las Redes Neuronales Artificiales . . . . .	47
3.9.2.	Redes neuronales convolucionales (convolutional neural network, CNN) . . . . .	50
3.9.3.	Aprendizaje por transferencia . . . . .	52
3.9.4.	Arquitecturas comunes aplicadas a imágenes de profundidad . . . . .	53
<b>4.</b>	<b>Desarrollo</b> . . . . .	<b>55</b>
4.1.	Descripción del esquema experimental . . . . .	57
4.1.1.	Configuración del sistema de medición . . . . .	57
4.1.2.	Adquisición de mediciones tras la implementación de la estructura metálica con superficie de cristal templado . . . . .	58
4.2.	Desarrollo del sistema de interfaz para adquisición de datos . . . . .	61
4.3.	Población muestra . . . . .	64
4.4.	Pruebas digitales de los métodos en la literatura . . . . .	65
4.4.1.	Índice de Staheli . . . . .	66
4.4.2.	Índice de Chippaux-Smirak . . . . .	68
4.5.	Preprocesamiento de las imágenes . . . . .	70
4.6.	Área bajo la curva . . . . .	73

4.7. Experimento 1: Clasificador Bayesiano con KDE sobre el Factor Standard	77
4.7.1. Desarrollo del método	77
4.7.1.1. Planteamiento general	77
4.7.1.2. Datos y particionado	78
4.7.1.3. Modelado KDE por clase	78
4.7.1.4. Regla bayesiana de decisión	78
4.8. Experimento 2: Truncado balanceado (13/13/13) y Bayes-KDE	79
4.8.1. Planteamiento y objetivo	79
4.8.2. Datos y particionado	80
4.8.3. Modelado KDE por clase	80
4.8.4. Regla bayesiana	81
4.9. Experimento 3: KDE + Bayes con clases 0 y 1	81
4.9.1. Planteamiento y objetivo	81
4.9.2. Distribución de datos	82
4.9.3. Estimación KDE por clase	82
4.9.4. Regla bayesiana de decisión	83
4.10. Experimento 4: MLP con doble particionado y particionado simple	83
4.10.1. Planteamiento y objetivo	83
4.10.2. Datos y particionado	84
4.10.3. Preprocesamiento y aplanamiento de matrices	84
4.10.4. Modelo MLP y parámetros de entrenamiento	85
4.11. Experimento 5: MLP sobre la matriz de profundidad de todo el pie	86
4.11.1. Planteamiento y objetivo	86
4.11.2. Datos y particionado	87
4.11.3. Modelo MLP y parámetros de entrenamiento	87
4.12. Experimento 6: Extracción de vectores de características	88
4.12.1. Planteamiento y objetivo	88
4.12.2. Datos y particionado	89
4.12.3. Modelos y parámetros de entrenamiento	89
4.13. Experimento 7: Matriz de profundidad + vectores de características (pie derecho real)	91
4.13.1. Planteamiento y objetivo	91
4.13.2. Datos y particionado	92

4.13.3. Modelos y parámetros de entrenamiento . . . . .	93
4.14. Experimento 8: vectores de características con balanceo mediante SMOTE . . . . .	94
4.14.1. Planteamiento y objetivo . . . . .	94
4.14.2. Datos y particionado . . . . .	95
4.14.3. Modelos y parámetros de entrenamiento . . . . .	96
<b>5. Resultados</b>	<b>98</b>
5.1. Resultados del Experimento 1 . . . . .	99
5.1.1. Densidades KDE por clase . . . . .	99
5.1.2. Desempeño en validación y prueba . . . . .	100
5.1.3. Métricas de desempeño . . . . .	101
5.2. Resultados del Experimento 2: Truncado 13/13/13 + Bayes-KDE .	101
5.2.1. Densidades KDE tras truncado . . . . .	101
5.2.2. Matrices de confusión en validación y prueba . . . . .	102
5.2.3. Métricas de desempeño . . . . .	104
5.3. Resultados del Experimento 3: KDE + Bayes con clases 0 y 1 . . . .	104
5.3.1. Densidades KDE en entrenamiento . . . . .	104
5.3.2. Matrices de confusión en validación y prueba . . . . .	105
5.3.2.1. Resultados en validación . . . . .	106
5.3.2.2. Resultados en prueba . . . . .	106
5.4. Resultados del Experimento 4: MLP con doble particionado y parti- cionado simple . . . . .	107
5.4.1. Comparación de métricas entre Partes 1 y 2 . . . . .	107
5.4.2. Desempeño detallado por parte . . . . .	107
5.4.2.1. Parte 1: doble particionado (train/val/test) . . . . .	107
5.4.2.2. Parte 2: particionado simple (train/test) . . . . .	108
5.5. Resultados del Experimento 5: MLP sobre la matriz de profundidad de todo el pie . . . . .	109
5.5.1. Curvas de pérdida y comportamiento del entrenamiento . . .	109
5.5.2. Métricas y comportamiento por parte . . . . .	112
5.6. Resultados del Experimento 6: Embeddings . . . . .	113
5.6.1. Métricas globales en validación . . . . .	113

---

5.6.2. Patrones en las matrices de confusión . . . . .	114
5.7. Resultados del Experimento 7: Matriz de profundidad + embeddings (pie derecho real) . . . . .	115
5.7.1. Historia de F1_weighted en la búsqueda de hiperparámetros .	115
5.7.2. Métricas y matrices de confusión en test . . . . .	117
5.8. Resultados del Experimento 8: Embeddings con balanceo mediante SMOTE . . . . .	119
5.8.1. Desempeño en validación (GridSearchCV) . . . . .	119
5.8.2. Métricas y matrices de confusión en prueba . . . . .	120
<b>6. Discusión</b>	<b>122</b>
<b>7. Conclusiones</b>	<b>125</b>
<b>Bibliografía</b>	<b>135</b>

---

## Índice de figuras

2.1. Diagrama del cuerpo libre del pie y arcos del pie [1]. . . . .	9
2.2. Tipos de pisadas [2]. . . . .	10
2.3. La primera columna desgloza el grupo de huesos que componen el tarso, metatarso y los falanges, por otra parte, la segunda columna abarca las tres secciones: retropié, mediopié y antepié, las cuales también son importantes de identificar y, por último, la tercera columna señala algunos tendones que son de interés, como lo es el tendón de Aquiles y el ligamento elástico [3]. . . . .	10
3.1. Clasificación de sensores con cámara de profundidad: activo (izquierda) y pasivo (derecha). . . . .	28
3.2. Profundidad estéreo. . . . .	30
3.3. Técnica ToF. . . . .	31
3.4. Componentes de la cámara. . . . .	32
3.5. Aplicación de la luz infrarroja en la piel. . . . .	33
3.6. Representación visual del problema de correspondencia. . . . .	34
3.7. Imagen real a nube de puntos. . . . .	37
3.8. Diferencias visuales para la comprensión de los términos profundidad de campo y campo de visión. . . . .	40
3.9. Ilustración de red convolucional (tomada de [4]). . . . .	50
4.1. Fotografía de la estructura física y representación digital del diseño desarrollado para la adquisición de imágenes de profundidad . . . . .	58
4.2. Visualización de puntos seleccionados manualmente sobre la imagen RGB. . . . .	61
4.3. Formulario 1 (izquierda) y Formulario 2 (derecha). . . . .	62
4.4. Configuración experimental para la captura de datos. . . . .	64
4.5. Visualización de segmentos requeridos para el cálculo del índice Staheli [5]. . . . .	67
4.6. Índice de Chippaux. . . . .	69
4.7. Histogramas por clase de pie. . . . .	74

4.8. Boxplot por clase de pie. . . . .	75
4.9. QQ plot por clase de pie. . . . .	75
5.1. Densidades KDE por clase con división estratificada. . . . .	100
5.2. Matrices de confusión en validación y prueba para el clasificador bayesiano con KDE. . . . .	100
5.3. Densidades KDE por clase (TRAIN) tras rebalanceo por truncado 13/13/13. . . . .	102
5.4. Matrices de confusión en validación y prueba con rebalanceo por truncado 13/13/13. . . . .	103
5.5. Densidades KDE por clase (TRAIN) para clases 0 y 1. . . . .	105
5.6. Matrices de confusión en validación y prueba para el Experimento 3. . . . .	105
5.7. Curva de pérdida del MLP en el Experimento 5(1) (tres clases, pie derecho). . . . .	110
5.8. Curva de pérdida del MLP en el Experimento 5(2) (dos clases, pie derecho). . . . .	110
5.9. Curva de pérdida del MLP en el Experimento 5(3) (tres clases, unificación de lado). . . . .	111
5.10. Curva de pérdida del MLP en el Experimento 5(4) (dos clases, unificación de lado). . . . .	111
5.11. Evolución del desempeño en F1_weighted para cada combinación de hiperparámetros evaluada en el Grid Search del Experimento 8. Las curvas permiten identificar la región de hiperparámetros que maximiza el desempeño para (a) Regresión Logística, (b) SVM lineal, (c) kNN y (d) Random Forest. . . . .	116
5.12. Comparación de métricas del Experimento 7 para los cuatro clasificadores evaluados. (a) Accuracy y (b) F1_macro se reportan como referencia, dado el desbalance de clases. (c) F1_weighted, métrica principal del experimento, muestra que la Regresión Logística obtiene el mejor desempeño global sobre el conjunto de prueba. . . . .	118

---

# Índice de tablas

3.1. Comparación entre imágenes de intensidad y de rango. . . . .	28
3.2. Comparación entre luz estructurada y luz codificada. . . . .	29
3.3. Características de sensores RGB e Imagen OmniVision OV9282 en Intel RealSense D435i. . . . .	42
3.4. Espacios de color comúnmente utilizados según la aplicación . . . . .	45
4.1. Clasificación del pie según el índice de Staheli . . . . .	67
4.2. Pruebas de normalidad aplicadas al Factor Standard en cada grupo. . . . .	73
4.3. Resultado de la prueba de Kruskal-Wallis aplicada al <i>Factor Standard</i> en los tres grupos de estudio. . . . .	76
4.4. Resultados de la prueba post hoc de Dunn con corrección de Bonferroni para el <i>Factor Standard</i> . . . . .	77
4.5. Distribución por clase antes y después del truncado, y por partición. . . . .	80
4.6. Distribución de clases en el conjunto de entrenamiento antes y después de SMOTE (Experimento 9). . . . .	96
5.1. Métricas de desempeño para validación y prueba. . . . .	101
5.2. Métricas de desempeño (macro-promedio) para el Experimento 2. . . . .	104
5.3. Comparación de métricas entre las Partes 1 (doble split 70/21/30) y 2 (split simple 70/30) del Experimento 4. . . . .	107
5.4. Matriz de confusión en test para el Experimento 6(4) (dos clases, unificación de lado). . . . .	113
5.5. Desempeño en validación sobre <i>embeddings</i> Dense-256 (40 imágenes). . . . .	114
5.6. Resultados de métricas de distintos modelos sobre <i>embeddings</i> Dense-256. . . . .	114
5.7. Desempeño en test ( $n = 30$ ) para el Experimento 8 sobre la representación combinada matriz+embeddings. . . . .	117
5.8. Desempeño en test sobre <i>embeddings</i> Dense-256 de pie derecho con balanceo SMOTE (30 imágenes). . . . .	120

---

---

## CAPÍTULO 1

---

# Introducción

### Contenido

---

<b>1.1. Justificación</b> . . . . .	<b>2</b>
<b>1.2. Objetivos</b> . . . . .	<b>3</b>
1.2.1. General . . . . .	3
1.2.2. Específicos . . . . .	3
<b>1.3. Hipótesis</b> . . . . .	<b>4</b>
<b>1.4. Productos obtenidos</b> . . . . .	<b>4</b>
<b>1.5. Organización</b> . . . . .	<b>5</b>

---

El pie plano es una alteración que tiene la característica de la ausencia del arco longitudinal medial del pie (ver figura 2.1). En México, el sobrepeso y la obesidad son un caso de estudio por parte del Fondo de las Naciones Unidas para la Infancia (UNICEF); estos factores están estrechamente relacionados al desarrollo del pie plano. Hasta ahora, entre el 15% y el 20% de la población padece esta alteración biomecánica. Los diagnósticos tradicionales se basan en la radiografía y huella plantar, pero estos enfoques tienen limitaciones en términos de precisión, accesibilidad y costo. La radiografía, aunque es precisa, requiere exposición a la radiación; por otro lado, los métodos basados en la huella plantar, como lo es la pedigráfica, son subjetivos y dependen en gran medida de la interpretación del especialista.

A partir de estas limitaciones, en el presente trabajo se plantea el desarrollo de una herramienta de apoyo al diagnóstico preventivo del pie plano utilizando imágenes de profundidad capturadas mediante una cámara no convencional. Dichas imágenes son procesadas con técnicas basadas en redes neuronales, con el propósito

de realizar una clasificación automática de la huella plantar. La intención no es sustituir el criterio clínico, sino ofrecer un recurso complementario que facilite la detección temprana, particularmente en entornos donde el acceso a especialistas es limitado.

Durante la pandemia, se puso de manifiesto la necesidad de herramientas que permitan realizar evaluaciones médicas sin necesidad de contacto físico. Es así como esta tecnología emergente no solo representa un avance en el diagnóstico del pie plano, sino que también sienta las bases para futuras aplicaciones en telemedicina, impulsando una atención médica accesible para poblaciones en zonas remotas.

## 1.1. Justificación

La UNICEF en su último reporte de julio del 2022, destacó a México como caso de estudio por sus altos porcentajes en sobrepeso para adultos, adolescentes y lamentablemente, en infantes también. Se puede observar que en edades que van desde los 5 a 19 años existe un 35 % de sobrepeso y obesidad [6]. Estos porcentajes resaltan que esto se convierte en un problema de salud pública y contribuye al desarrollo de alteraciones biomecánicas, así como complicaciones a largo plazo en la calidad de vida [7]. Por otra parte, el blog de la Secretaría de Salud del Gobierno de México, en su última actualización reportó en septiembre de 2016 que entre el 15 % y 20 % de la población padece pie plano [8]. Tanto el sobrepeso como el pie plano prevalecen en proporciones significativas en la población de México, por lo que es crítico desarrollar herramientas que utilicen la tecnología para ayudar al especialista en el diagnóstico temprano de pie plano en poblaciones con altos índices de masa corporal (IMC). Por esta razón, es importante considerar que la estructura del pie está estrechamente relacionada con los huesos de la pierna y el muslo, lo que afecta su funcionalidad. A su vez, esto puede afectar las alineaciones del tobillo, pierna, pelvis y columna vertebral [9], empezando por el tobillo. Todos estos datos dan indicio a que esta patología no debe ser subestimada, ya que su detección temprana

permitiría prevenir desequilibrios posturales y funcionales que pueden agravarse con el tiempo, especialmente en poblaciones con sobrepeso.

## 1.2. Objetivos

### 1.2.1. General

Desarrollar y evaluar una metodología de análisis de huellas plantares basada en imágenes de profundidad y técnicas de aprendizaje automático para apoyar el diagnóstico preventivo del pie plano, integrando mediciones morfológicas y representaciones latentes aprendidas mediante redes neuronales.

### 1.2.2. Específicos

Para cumplir con el objetivo general de esta propuesta, se han planteado los siguientes objetivos específicos:

- Analizar los criterios clínicos tradicionales para el diagnóstico de pie plano e integrar los índices Staheli y Chippaux-Smirak como referencia comparativa.
- Diseñar un protocolo de adquisición que permita obtener imágenes en el espacio de color rojo-verde-azul con profundidad (Red, Green, Blue, Depth, RGB-D) de huellas plantares con profundidad corregida y adecuada segmentación.
- Implementar un pipeline para calcular métricas morfológicas (área, volumen, altura promedio bajo el arco) a partir de las imágenes de profundidad procesadas.
- Extraer embeddings mediante una red neuronal convolucional preentrenada y evaluar modelos clásicos de clasificación sobre dichos vectores.

- Comparar el desempeño de métricas morfológicas, embeddings y métodos tradicionales para determinar su capacidad discriminativa entre pie plano y pie no plano.

### 1.3. Hipótesis

Los descriptores obtenidos a partir de imágenes de profundidad, combinados con representaciones latentes extraídas mediante redes neuronales, permiten identificar patrones morfológicos asociados al pie plano, mostrando potencial como herramienta complementaria para su diagnóstico preventivo.

### 1.4. Productos obtenidos

Durante el desarrollo de esta investigación se generaron productos académicos que evidencian el avance, la pertinencia y la validación científica del proyecto por parte de la comunidad nacional. Entre ellos se encuentran:

- **Póster aceptado como primera autora** en el *Congreso Nacional de Física*, realizado del 12 al 17 de octubre de 2025 en Toluca, México. En este trabajo se presentaron los avances preliminares del sistema de análisis basado en imágenes de profundidad para el diagnóstico preventivo del pie plano.
- **Póster aceptado como primera autora** en el *Congreso de Ciencia, Tecnología e Innovación 2025*, efectuado del 28 al 30 de octubre en Oaxaca, México. En esta presentación se expusieron los resultados iniciales del protocolo de adquisición, el procesamiento de imágenes de profundidad y la extracción de características morfológicas del arco plantar.

- **Repositorio personal de datos de imágenes de profundidad**, generado durante el proceso experimental de esta investigación. Este dataset incluye mapas de profundidad en formato CSV e imágenes RGB asociadas, obtenidos mediante la cámara Intel RealSense. El conjunto de datos será puesto a disposición pública para fines académicos y de investigación en el siguiente enlace: <https://drive.google.com/drive/folders/1P0Lcu8gV5Z72uQnzkd49xJM0ze23ULn1?usp=sharing>.
- **Repositorio personal de software del sistema desarrollado**, el cual incluye los scripts implementados en Python para la adquisición de datos, el preprocesamiento de imágenes de profundidad, el cálculo de métricas morfológicas y la implementación de los modelos de clasificación. Este repositorio será compartido para facilitar su consulta y reutilización académica en el siguiente enlace: <https://drive.google.com/drive/folders/1P0Lcu8gV5Z72uQnzkd49xJM0ze23ULn1?usp=sharing>.
- **Manuscrito de divulgación científica** titulado (*Lo que no deja huella también importa: detección del pie plano a partir de su forma*), enviado a la revista *Komputer Sapiens*, en el cual se presenta una descripción accesible del sistema desarrollado y sus posibles aplicaciones en el diagnóstico preventivo del pie plano.

Estos productos representan la difusión temprana de los resultados y la consolidación del trabajo como una propuesta viable dentro del campo de la visión por computadora aplicada al análisis biomédico.

## 1.5. Organización

El presente trabajo de tesis se estructura en seis capítulos, cuyo contenido se describe a continuación.

En el **Capítulo 2** se presentan los antecedentes y el estado del arte relacionados con el diagnóstico del pie plano. Se revisan las definiciones clínicas fundamentales, los métodos utilizados en la práctica médica, así como enfoques basados en análisis morfológico 2D, presión plantar e imágenes tridimensionales. Finalmente, se discuten las limitaciones de los métodos existentes y la brecha de conocimiento que motiva esta investigación.

El **Capítulo 3** desarrolla el marco teórico necesario para comprender la metodología propuesta. Se abordan los principios físicos de la visión por computadora con profundidad, la obtención de mapas de profundidad, la operación de la cámara Intel RealSense D435i, y los conceptos de segmentación por color, redes neuronales, *transfer learning* y técnicas comunes aplicadas al procesamiento de datos RGB-D.

En el **Capítulo 4** se describe a detalle el desarrollo experimental. Este capítulo incluye la configuración del sistema de medición, el protocolo de adquisición con la estructura metálica y superficie de vidrio templado, el procesamiento de imágenes, el cálculo de métricas morfológicas y la construcción de múltiples experimentos de clasificación. Se presentan distintos enfoques, desde índices clínicos tradicionales hasta modelos supervisados basados en matrices de profundidad, *embeddings* y técnicas de balanceo como SMOTE.

El **Capítulo 5** expone los resultados obtenidos en cada experimento. Se analizan las distribuciones de las características morfológicas, el desempeño de clasificadores bayesianos, perceptrón multicapa (multilayer perceptron, MLP) y modelos que utilizan como entrada *vectores de características*, así como las métricas globales, las matrices de confusión y el efecto del balanceo de clases. Este capítulo sintetiza la evidencia experimental que sustenta la validación de la metodología propuesta.

En el **Capítulo 6** se presenta la discusión general del trabajo. A partir de los resultados experimentales, se integran las observaciones más relevantes, se analizan las fortalezas y limitaciones de cada enfoque y se reflexiona sobre los elementos que influyen en la capacidad de los modelos para identificar el pie plano.

Finalmente, en el **Capítulo 7** se exponen las conclusiones del estudio. Se destacan las aportaciones principales, se vinculan los hallazgos con las preguntas de investigación planteadas en la introducción y se señalan posibles líneas de trabajo futuro.

---

---

## CAPÍTULO 2

---

# Antecedentes y Estado del Arte

### Contenido

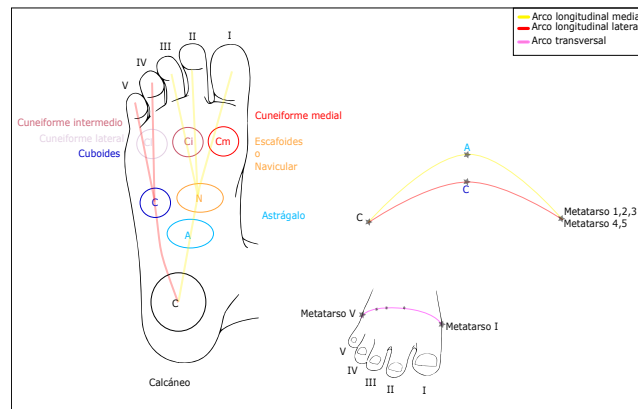
---

<b>2.1. Conceptos y definiciones clínicas del pie . . . . .</b>	<b>8</b>
<b>2.2. Métodos diagnósticos en la práctica clínica . . . . .</b>	<b>11</b>
<b>2.3. Métodos computacionales aplicados al diagnóstico del pie plano . . . . .</b>	<b>17</b>
2.3.1. Clasificación basada en el análisis morfológico 2D . . . . .	17
2.3.2. Clasificación basada en datos de presión plantar . . . . .	20
2.3.3. Modelado 3D y Escaneo Tridimensional: La Base de los Enfoques de Vanguardia . . . . .	21
<b>2.4. Limitaciones de los enfoques revisados . . . . .</b>	<b>23</b>
<b>2.5. Brecha encontrada . . . . .</b>	<b>23</b>

---

### 2.1. Conceptos y definiciones clínicas del pie

Un pie plano se caracteriza por la altura de su arco, si este es pequeño, se categoriza como pie plano, si este está muy pronunciado, se caracteriza como pie cavo. La altura del arco longitudinal medial es la que orienta en la investigación del pie plano (ver Figura 2.1) [10]. El pie plano flexible suele tener un aspecto normal cuando está en reposo, es decir, que no está soportando carga alguna, pero al soportarla, su apariencia luce como si no estuviese ahí el arco medial. Por otra parte, en el pie plano rígido, destaca su planicidad incluso sin carga alguna [11], [12].

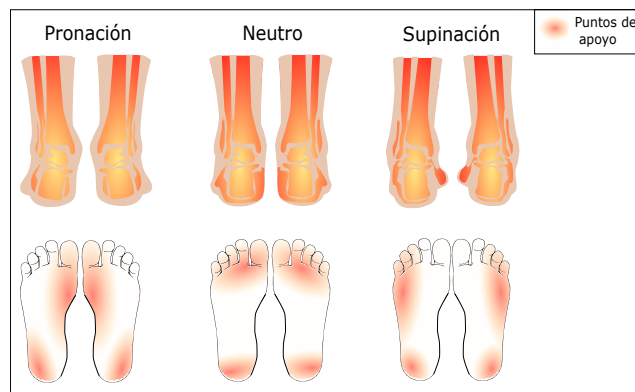


**Figura 2.1:** Diagrama del cuerpo libre del pie y arcos del pie [1].

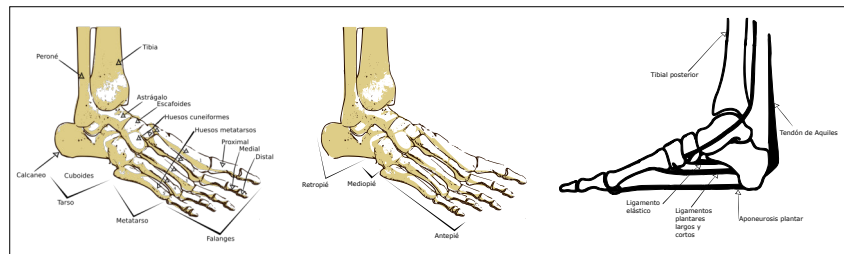
Sin embargo, el estudio de estas estructuras es fundamental para comprender las alteraciones biomecánicas [13] que se presentan en el pie, por lo que amerita atención médica y exploración. Además, existen algunos factores como por ejemplo el índice de masa corporal que está estrechamente relacionado al pie plano [14].

Debido a la altura del arco longitudinal medial este puede propiciar que el tipo de pisada sea pronador o supinador. Pronador es cuando el pie ejerce mayor presión en la parte interna de este, haciendo que la proyección de la huella plantar sea hacia afuera; en el caso del supinador, esta pisada ejerce mayor presión en la parte externa del pie, haciendo que la proyección de la huella plantar sea hacia afuera (ver Figura 2.2) [2]. Finalmente esto puede desencadenarse en inflamación de las articulaciones, dolor de talón. Incluso, análisis médicos mencionan que la tibia anterior se encuentra comprometida debido a esta deficiencia provocando así periostitis tibial. El pie plano puede ser consecuencia de varias condiciones subyacentes, como la disfunción del tendón tibial posterior, los cambios de Charcot en el mediopié o la artritis reumatoide. Aunque estas alteraciones son las verdaderas causantes de la deformidad, el pie plano suele ser la manifestación más evidente a simple vista, lo que facilita su detección inicial. Sin embargo, este signo visible puede ser solo la punta del iceberg, ya que estas condiciones afectan significativamente la estructura y función del mediopié y el retropié (ver Figura 2.3, centro), causando un impacto progresivo y, a menudo, debilitante [15]. Si bien el pie plano puede ser adquirido como lo es en la

adulthood debido a varias anomalías antes mencionadas, el pie plano también puede presentarse como una anomalía congénita, es decir, desde el nacimiento del niño. Esto se ve reflejado comúnmente en niños y dado que la mayoría de las veces es asintomático, es por ello que es un área importante de explorar para poder ser detectada a tiempo [16]. Los métodos tradicionales requieren de un especialista y esto es difícil ya que no siempre los hay en entornos de atención primaria.



**Figura 2.2:** Tipos de pisadas [2].



**Figura 2.3:** La primera columna desglosa el grupo de huesos que componen el tarso, metatarso y los falanges, por otra parte, la segunda columna abarca las tres secciones: retropié, mediopié y antepié, las cuales también son importantes de identificar y, por último, la tercera columna señala algunos tendones que son de interés, como lo es el tendón de Aquiles y el ligamento elástico [3].

Existen métodos directos e indirectos, los métodos indirectos usan la huella y la fotografía, mientras que los métodos directos usan técnicas antropométricas y la radiografía. La diferencia principal radica en la precisión y la objetividad de las me-

diciones: los métodos directos proporcionan datos más confiables sobre la estructura ósea, mientras que los métodos indirectos dependen de interpretaciones que pueden ser menos precisas. El contenido a continuación está basado en la literatura existente acerca de cómo abordan el diagnóstico del pie plano, desde una perspectiva médica y de procesamiento computacional, destacando sus ventajas y mostrando sus debilidades, lo cual justifica la necesidad de un nuevo avance en esta línea de investigación.

## 2.2. Métodos diagnósticos en la práctica clínica

En el ámbito médico, el Instituto Mexicano del Seguro Social (IMSS) [17] desarrolló una guía de práctica clínica que establece directrices para el diagnóstico y manejo del pie plano en población mexicana. Este documento enfatiza el uso de pruebas específicas como Jack's Great toe y la Prueba de Rodríguez Fonseca para evaluar la flexibilidad del arco plantar y clasificar el tipo de pie plano. La primera se realiza con el paciente en bipedestación mediante la dorsiflexión pasiva del hallux (dedo gordo del pie); la ausencia de elevación del arco medial durante esta maniobra indica incompetencia ligamentaria. De forma complementaria, la prueba de Rodríguez Fonseca, basada en la elevación del paciente sobre las puntillas, permite analizar la estabilidad funcional del arco longitudinal medial. Durante esta maniobra, el pie plano flexible suele mostrar la reaparición del arco plantar acompañada de la corrección del talón desde una posición en valgo (apoyo predominante hacia el borde interno del pie) hacia varo (desplazamiento del apoyo hacia el borde externo), lo que refleja una adecuada capacidad de adaptación biomecánica del pie bajo carga [18]. En contraste, la persistencia del valgo del retropié y la ausencia de elevación del arco durante la ejecución de la prueba sugieren la presencia de un pie plano rígido o disfunción del tibial posterior. Si bien facilita el diagnóstico en atención primaria y evita intervenciones innecesarias, tiene aplicación limitada y omite el uso de nuevas herramientas tecnológicas.

A nivel internacional varios estudios han evaluado la limitada fiabilidad de las mediciones de la altura del arco longitudinal medial del pie basadas en el análisis de huellas plantares. El estudio [19] comparó los resultados de las huellas plantares con mediciones radiográficas, en 38 niños con pie plano flexible (28 niños y 10 niñas), con edades entre los 3.7-11.7 años y una media de 6.4 años. Se concluyó que la precisión de las huellas plantares es insuficiente, lo anterior debido a la influencia de los tejidos blandos, que impiden obtener medidas específicas y consistentes de la huella. Mientras que en [20] se llevó a cabo la evaluación de la altura del arco longitudinal medial mediante radiografías y el índice del arco a partir de la huella plantar. Se evaluó la forma del pie mediante el uso de un dispositivo que registra la presión plantar y radiografías. El estudio incluyó a 100 voluntarios (24 hombres y 76 mujeres), con edades entre los 12 y 85 años, que presentaban molestias en el pie, excluyendo a quienes habían tenido cirugías previas. Se midieron datos como altura y peso para calcular el IMC, y se compararon los resultados del dispositivo con los ángulos obtenidos en las radiografías. Aunque no se encontró una correlación fuerte con uno de los ángulos medidos, el estudio sugiere que mejorar la automatización del proceso podría ser útil para evaluaciones de salud rutinarias.

En la revisión sistemática llevada a cabo en [21] se analizaron diferentes definiciones y métodos empleados para caracterizar el pie plano en población pediátrica. A partir de un conjunto amplio de estudios, los autores identificaron que, pese a la diversidad de enfoques, los índices de Chippaux-Smirak y Staheli son los que aparecen con mayor frecuencia en la literatura y en la práctica clínica. Esta recurrencia ha permitido que ambos métodos se consoliden como referentes en la evaluación de la huella plantar y la determinación del arco longitudinal del pie. No obstante, la revisión también pone en evidencia que, a pesar de su popularidad, existen variaciones en los resultados obtenidos con estos índices, lo que limita la posibilidad de establecer una estandarización sólida y resalta la necesidad de seguir explorando métodos de diagnóstico más consistentes.

En este marco, la literatura ha explorado tanto poblaciones pediátricas como adultas para contrastar la utilidad de estos índices. Un ejemplo representativo es el siguiente estudio [22], en el que se analizaron 335 niños de entre 6 y 10 años de la

región de Anatolia Central mediante el método de pedigrafía. Esta técnica, basada en la obtención de huellas plantares con láminas entintadas y reticuladas, permitió obtener impresiones estáticas en carga que posteriormente se sometieron a un análisis geométrico. A partir de dichas huellas fue posible calcular hasta 18 parámetros distintos, entre ellos longitudes (total y truncada del pie), circunferencias, medidas oblicuas y, de manera destacada, los índices de Staheli y Chippaux-Smirak. Estos últimos se obtuvieron mediante algoritmos sencillos de proporciones lineales: el índice de Chippaux-Smirak como la razón entre el ancho mínimo del mediopié y el ancho máximo del antepié multiplicado por cien, y el índice de Staheli como la relación entre el segmento más angosto del mediopié y el retropié. El valor añadido de este procedimiento es que permitió generar mediciones objetivas y comparables que no dependen de la observación clínica directa, proporcionando un marco más confiable para el análisis de la morfología plantar en población infantil.

Los hallazgos revelaron que, a medida que aumenta la edad, los valores de Staheli y Chippaux tienden a disminuir, reflejando un fortalecimiento progresivo del arco medial. Según el índice de Staheli, la prevalencia fue de 63.3% para pie plano, 27.7% para arco normal y 9.8% para pie cavo, lo que evidencia la alta frecuencia de pies planos en edades tempranas y respalda la noción de que muchos casos son flexibles y mejoran con el desarrollo. Este resultado posiciona al índice de Staheli como el más empleado y de mayor peso en la clasificación final, aunque el índice de Chippaux-Smirak también fue relevante al compararse con estudios internacionales en Brasil, Alemania y el Congo, mostrando su utilidad para la estandarización. Entre las limitaciones, los autores reconocen que la muestra se restringió a una sola región y que el uso de huellas entintadas, aunque práctico, depende de la correcta aplicación de carga sobre el pie y no ofrece la precisión de tecnologías tridimensionales más modernas.

Si bien el trabajo mencionado aporta información en varias dimensiones antropométricas, lo más significativo en el contexto del pie plano es la manera en que confirma la vigencia de los índices de Staheli y Chippaux-Smirak en la práctica clínica y en la investigación internacional. El primero emerge como el criterio de referencia en la clasificación, mientras que el segundo, aunque menos central en las

conclusiones, complementa y refuerza el análisis al permitir comparaciones con otras poblaciones. En conjunto, este estudio demuestra cómo la aplicación de mediciones objetivas basadas en algoritmos simples contribuye a un diagnóstico más confiable en población pediátrica, al tiempo que pone de relieve la necesidad de metodologías estandarizadas y del aprovechamiento de tecnologías digitales para superar las limitaciones de la pedigrafía tradicional.

De manera complementaria, investigaciones en población adulta han comparado directamente el índice de Chippaux-Smirak con parámetros estructurales, como el ángulo de Clarke, con el fin de valorar su concordancia diagnóstica. Un ejemplo ilustrativo es el estudio de González-Martín et al. (2017) [23], que analizó huellas plantares estáticas obtenidas con pedigráfico de tinta.

En este estudio, el Índice de Chippaux-Smirak (Chippaux-Smirak Index, CSI) y el ángulo de Clarke fueron evaluados mediante un método de pedigráfico con tinta, técnica basada en la obtención de huellas plantares estáticas. Para el cálculo del CSI, el proceso incluyó pasos detallados: primero, se trazó una línea en la parte más prominente del antepié y otra en el retropié, definiendo los límites anterior y posterior de la huella. Luego, se identificaron los puntos más estrechos en la región del mediopié, correspondiente al arco longitudinal medial. Se colocaron dos puntos en estas constricciones: uno en el borde interno y otro en el externo de la huella, midiéndose la distancia entre ellos para obtener el ancho del arco. Paralelamente, se marcaron dos puntos en las zonas más anchas del antepié (a nivel de las cabezas metatarsales), midiéndose la distancia entre ellos para obtener el ancho del antepié. Estos valores se incorporaron a la fórmula:

$$CSI = \frac{\text{Ancho del antepié}}{\text{Ancho del arco}} \times 100 \quad (2.1)$$

Clasificándose como pie plano valores superiores al 45 %.

Para el ángulo de Clarke, se siguió la siguiente metodología: se trazó una línea base que uniera los puntos más prominentes del antepié y el retropié en el borde

interno de la huella. Luego, se identificó el punto de máxima concavidad del arco longitudinal medial (punto donde la curva comienza a ascender hacia el antepié). Desde este punto de concavidad, se trazó una segunda línea hasta el punto más prominente del antepié en el borde medial. Utilizando un transportador, se midió el ángulo formado entre estas dos líneas, considerándose pie plano valores  $\leq 30^\circ$  y pie cavo valores  $> 45^\circ$ , con el rango normal entre  $31^\circ - 45^\circ$ . Este enfoque permitió cuantificar la prevalencia de pie plano en una muestra de 1,002 adultos, aunque la subjetividad en la delimitación de puntos anatómicos introdujo limitaciones en la reproducibilidad.

Los resultados revelaron una discordancia significativa entre ambos métodos. Mientras el CSI identificó una prevalencia de pie plano del 62.0%, el ángulo de Clarke sólo detectó un 29.7%, con una concordancia baja (coeficiente Kappa: 0.21–0.33) y una correlación intraclase negativa (intraclass correlation coefficient, ICC  $\approx -0.43$ ). El ángulo de Clarke mostró una sensibilidad limitada (51.1%) pero una especificidad aceptable (75.0%) al usar el CSI como referencia. Factores como la edad avanzada y el elevado IMC influyeron en estas discrepancias; por ejemplo, un IMC mayor redujo la probabilidad de discordancia (odds ratio, OR: 0.969), posiblemente debido a que el CSI es más susceptible a cambios en la morfología plantar por tejido adiposo, mientras que el ángulo de Clarke depende de puntos óseos más estables.

El estudio concluye que, aunque el CSI es más sensible para detectar alteraciones del arco longitudinal, especialmente en poblaciones envejecidas o con obesidad, su tendencia a sobrestimar el pie plano requiere complementarse con métodos estructurales como el ángulo de Clarke para evitar diagnósticos falsos. La falta de estandarización en las técnicas de medición y los puntos de corte subraya la necesidad de protocolos unificados y herramientas digitalizadas que minimicen la variabilidad interobservador. Estos hallazgos son cruciales para la práctica podológica, destacando que la elección del método diagnóstico debe considerar el contexto clínico, las características del paciente y la posible integración de tecnologías emergentes para mejorar la precisión en la evaluación del pie plano.

En México, la revisión [24] amplía el panorama al analizar diferentes patologías ortopédicas y su relación con la presión plantar. En el caso específico del pie plano, se destaca que la distribución del peso tiende a ser uniforme, a diferencia del pie cavo, donde la carga se concentra en el arco medial. Las plataformas de presión son herramientas esenciales en biomecánica que permiten analizar la distribución de carga en los pies. Estos sistemas, compuestos por matrices de sensores, capturan datos de fuerza en tiempo real durante la marcha o la postura. El estudio [25] es fundamental porque confirma que la morfología del pie plano, específicamente la falta de arco, altera de manera significativa su función biomecánica. El análisis con la plataforma demostró que los pies planos, incluso los asintomáticos, tienen una carga anormalmente alta en la región del mediopié (arco medial) y una redistribución de la presión hacia la parte interior del antepié. Esto estableció una conexión directa entre la estructura del pie y su función, sirviendo como base para futuras investigaciones. La verdadera utilidad de las plataformas de presión reside en las ecuaciones de análisis que permiten interpretar los datos de forma objetiva y reproducible. Dos de las más importantes son el índice de excursión del centro de presión (CPEI) y el índice de valgo del maléolo (MVI). El CPEI mide la desviación lateral del centro de presión durante el movimiento. Un valor bajo indica una pronación excesiva, común en el pie plano flexible, donde el peso no sigue su trayectoria normal. Por otro lado, el MVI evalúa la alineación estática del tobillo, o retropié, basándose en la posición de los maléolos. Un MVI positivo es un indicador clave de una alineación en valgo, característica del pie plano. Estas ecuaciones convierten los complejos patrones de la marcha y la estructura ósea en valores numéricos concretos, lo que facilita el diagnóstico y seguimiento de esta condición.

Más allá del diagnóstico, las plataformas de presión son herramientas cruciales para evaluar la eficacia de las intervenciones terapéuticas. Un segundo estudio [26] relevante analizó el efecto de la terapia de rehabilitación en niños con pie plano. Usando un sistema de plataforma Emed, los investigadores midieron la distribución de la presión plantar y el área de contacto en el mediopié antes y después de la rehabilitación. Los resultados mostraron un aumento de la presión media y una disminución del área de contacto en el mediopié en todos los participantes. Esta disminución del área de contacto sugiere que la altura del arco del pie aumentó,

lo que podría deberse a la activación de los músculos estabilizadores del pie. Este hallazgo, aunque preliminar por ser un estudio de caso, indica que las plataformas de presión son útiles para rastrear los cambios en la función estática del pie y evidenciar el efecto de la rehabilitación en el tratamiento del pie plano infantil.

A pesar de su valor, las plataformas de presión presentan limitaciones importantes. La principal es que la evaluación se realiza en un entorno de laboratorio, lo que puede no replicar la biomecánica natural del pie en el día a día o dentro de un calzado. Además, la resolución espacial de los sensores (por ejemplo, 25 mm<sup>2</sup> por sensor en el estudio de Ledoux) puede ser insuficiente para delinear con precisión la geometría ósea. El alto costo y la falta de portabilidad también limitan su uso clínico generalizado. Sin embargo, estas limitaciones han impulsado la investigación hacia el desarrollo de herramientas más accesibles y prácticas, como el calzado instrumentado, con el objetivo de continuar el análisis de la función del pie fuera del laboratorio y permitir una prevención y tratamiento más personalizados.

## 2.3. Métodos computacionales aplicados al diagnóstico del pie plano

### 2.3.1. Clasificación basada en el análisis morfológico 2D

Entre los métodos computacionales, el análisis de imágenes 2D destaca por su balance entre accesibilidad y objetividad. Este enfoque se centra en digitalizar y procesar la huella plantar para cuantificar parámetros que tradicionalmente se evalúan de forma visual, como la altura y la solidez del arco longitudinal medial, clave para diagnosticar el pie plano.

El trabajo [27] es un ejemplo seminal de esta familia. Su principal contribución fue el desarrollo de un sistema de adquisición que utiliza un escáner electrónico para capturar la huella bajo carga, obteniendo una imagen digital de alta fidelidad. Para

el procesamiento en MATLAB, el autor implementó un algoritmo que se basó en la Transformada de Hough, una técnica robusta para la detección de patrones y formas geométricas dentro de la imagen. Este método permitió identificar y medir con precisión las zonas de contacto y no contacto de la huella, superando la subjetividad de la inspección visual. El sistema demostró ser eficaz para calcular el índice del arco plantar de manera automatizada, proporcionando una medida objetiva, reproducible y confiable para diferenciar entre un pie normal y un pie plano. El rendimiento del sistema se validó al mostrar una alta consistencia en las mediciones, lo que lo posiciona como una herramienta valiosa para un pre-diagnóstico objetivo.

Los hallazgos del estudio confirmaron que una técnica digital, apoyada por algoritmos de procesamiento de imágenes, reduce significativamente la variabilidad inherente al diagnóstico puramente visual. Sin embargo, el trabajo también identificó limitaciones importantes. El sistema depende de un equipo de escaneo específico que requiere una calibración cuidadosa para garantizar la precisión de las mediciones. Como trabajo futuro, el autor sugirió expandir la base de datos de huellas y explorar algoritmos más avanzados para mejorar aún más la generalización y robustez del sistema en la detección del pie plano.

El trabajo de [28] desarrolló un podoscopio digital basado en Raspberry Pi para automatizar el diagnóstico de pie plano. El algoritmo implementado, que aplica un pipeline de procesamiento de imágenes que incluye escala de grises, operaciones morfológicas y binarización de Otsu, calcula de manera objetiva el porcentaje de ausencia del arco plantar. Esta métrica cuantitativa permitió clasificar las huellas en categorías específicas de normales, planos (grados 1-4) o cavos (grados 1-2). El sistema demostró un alto rendimiento, alcanzando una exactitud del 97.28% en comparación con las valoraciones de especialistas, lo que lo posiciona como una herramienta confiable para un diagnóstico preliminar objetivo.

La principal contribución de este estudio radica en proveer un método rápido (aproximadamente 105 segundos por paciente) y de bajo costo que reduce significativamente la subjetividad inherente a la evaluación visual tradicional del pie plano. No obstante, el enfoque presenta limitaciones. La efectividad del algoritmo depen-

de críticamente de condiciones de iluminación controlada durante la captura de la imagen, lo que puede restringir su uso en entornos no especializados. Asimismo, la validación con una muestra de 184 niños sugiere la necesidad de evaluar su generalización en poblaciones más amplias y diversas.

Como trabajo a futuro, el propio estudio [28] sugiere la ampliación de la base de datos de entrenamiento y la integración de técnicas de aprendizaje profundo. Esta evolución podría permitir que el sistema capture patrones morfológicos más complejos, mejorando su robustez y autonomía para manejar la variabilidad natural entre pacientes y diferentes condiciones de captura.

El estudio de [29] adoptó un enfoque de aprendizaje automático clásico para la clasificación de alteraciones del pie, incluido el pie plano. A partir de un conjunto de 200 imágenes de huellas plantares, se extrajo un vector de 11 características morfológicas que cuantificaban geoméricamente la forma y la estructura del arco plantar. Este vector de características fue utilizado para entrenar y comparar cinco algoritmos distintos: árbol de decisión (decision tree, DT), bosques aleatorios (random forest, RF), máquinas de vectores de soporte (support vector machines, SVM), redes neuronales artificiales (artificial neural networks, ANN) y XGBoost. El Árbol de Decisión demostró el mejor rendimiento, alcanzando una precisión de clasificación del 95 % para distinguir entre pies normales y pies con alteraciones como el pie plano, superando a los demás modelos evaluados.

La principal contribución de este trabajo reside en demostrar la utilidad de las características morfológicas extraídas de forma manual como predictores robustos para algoritmos de machine learning, proporcionando un método rápido y objetivo para el screening de pie plano. Los resultados indican que incluso modelos relativamente simples pueden lograr una alta precisión cuando se alimentan con parámetros morfológicos bien seleccionados, ofreciendo una alternativa eficaz a la evaluación visual subjetiva.

Sin embargo, la investigación presentó limitaciones significativas. La efectividad del modelo depende críticamente de la calidad y estandarización en la captura de

las imágenes iniciales para una correcta extracción de características. Además, el tamaño reducido de la muestra (200 imágenes) y la falta de especificación sobre la demografía de los participantes plantean dudas sobre la generalización de los resultados a poblaciones más amplias y diversas. Los autores sugieren que futuras investigaciones exploren el uso de redes neuronales convolucionales (Convolutional Neural Network, CNN) para automatizar la extracción de características y capturar patrones más complejos, así como ampliar el conjunto de datos para mejorar la robustez del modelo y su aplicabilidad clínica en el diagnóstico del pie plano

### 2.3.2. Clasificación basada en datos de presión plantar

El estudio de [30] se centró en el análisis cinético, utilizando datos de presión plantar obtenidos de una plataforma baropodométrica (FreeMed®) para clasificar alteraciones como el pie plano y el pie cavo. El objetivo principal fue comparar el rendimiento de dos técnicas de inteligencia computacional: Mapas cognitivos difusos (Fuzzy Cognitive Maps, FCM) entrenados con un Algoritmo Genético (Genetic Algorithm, GA) y una Red Neuronal Multicapa (Multilayer Perceptron Neural Network, MLPNN), utilizando un conjunto de datos relativamente pequeño. La medida objetiva y confiable que proporciona este sistema es el porcentaje de carga distribuido en 14 áreas específicas del pie (dedos, cabezas metatarsales, mediopie y talón), normalizado según el peso corporal total del paciente. Este enfoque permite una evaluación funcional que va más allá de la morfología, capturando cómo se distribuye la presión durante la marcha.

El algoritmo central para la clasificación fue FCM, cuya matriz de pesos fue optimizada mediante un Algoritmo Genético. Este enfoque “fuzzy” se comparó con una MLPNN tradicional. El rendimiento del modelo GA-FCM fue superior, alcanzando una precisión del 91.17% en la clasificación de los datos de prueba, superando a la MLPNN (87.29%). El sistema demostró una alta capacidad para detectar la alteración cuando estaba presente (sensibilidad del 94%) y para confirmar su ausencia (especificidad del 87%). Los hallazgos confirmaron que los modelos difusos, que emulan el razonamiento humano, son especialmente efectivos para manejar la

incertidumbre y los conjuntos de datos pequeños, proporcionando una herramienta de apoyo al diagnóstico comprensible y con un alto desempeño.

A pesar del alto rendimiento, el estudio presenta limitaciones. La principal es el tamaño de la muestra (151 pacientes), que, aunque suficiente para demostrar la efectividad del método, no es lo suficientemente amplia para ser implementada directamente en la práctica clínica real. Además, la construcción del modelo FCM depende de un proceso de optimización mediante GA que puede ser computacionalmente demandante. Los autores sugieren que trabajos futuros se centren en ampliar la base de datos de presión plantar y explorar otros algoritmos de aprendizaje para los FCM, como los basados en métodos hebbianos o algoritmos bacterianos, para mejorar el tiempo de convergencia y la robustez del modelo en la detección del pie plano basado en su función.

### 2.3.3. Modelado 3D y Escaneo Tridimensional: La Base de los Enfoques de Vanguardia

El estudio revisado [31] se centró en la caracterización tridimensional del arco medial del pie en población infantil y adolescente, con el propósito de comprender su desarrollo y establecer parámetros objetivos que permitieran diferenciar entre pies planos, normales y cavos. Para ello se empleó un sistema de captura 3D de alta precisión (UPod), cuyo costo ronda los 5000 dólares estadounidenses, lo que lo convierte en un equipo costoso y poco accesible para entornos clínicos convencionales. Este escáner permitió obtener con gran detalle la longitud del arco medial (arch length, AL), el punto medio de dicha longitud (medial arch length midpoint, MALM) y la altura del arco medial (medial arch height, MAH). El procedimiento consistió en medir la AL desde el centro del talón hasta la cabeza del segundo metatarsiano; posteriormente, se calculó su punto medio y desde allí se registró la altura vertical hasta la piel en contacto con el suelo. La estandarización se buscó a través del uso de estos puntos anatómicos y de índices proporcionales, como la razón de la altura del arco (arch height ratio, AHR), que elimina variaciones debidas al crecimiento y permite comparaciones confiables entre distintos sujetos y edades.

Una vez obtenidos estos parámetros, los autores realizaron un análisis estadístico exhaustivo de correlaciones entre múltiples variables, como edad, sexo, índice de masa corporal y los distintos índices estructurales del pie. Aunque se presentaron numerosos gráficos, no todas las relaciones fueron significativas; en la mayoría de los casos se observaron correlaciones débiles o inconsistentes. La excepción fue la asociación entre el índice del arco (arch index, AI), derivado de la proporción de la huella plantar, y el AHR. Esta relación resultó fuerte y negativa, lo cual es clínicamente esperable: a mayor AI, el arco es más bajo, mientras que un mayor AHR refleja un arco más elevado. A partir de esta asociación se estableció un modelo de regresión lineal simple, en el que el AI fue la variable dependiente y el AHR la independiente, obteniendo la siguiente ecuación:

$$AI = 0.318 - (0.17 \cdot AHR) \quad (2.2)$$

El coeficiente de determinación ( $R^2 = 0.483$ ) indicó que aproximadamente el 48.3% de la variabilidad del AI puede predecirse a partir del AHR. Esto es relevante porque el AHR, al calcularse directamente con medidas de altura y longitud del arco, resulta más sencillo de obtener en la práctica clínica que el AI, que depende de pedigrama o de imágenes de huella plantar. De esta manera, el estudio sugiere un atajo práctico que facilita la clasificación del pie plano sin necesidad de métodos invasivos o equipos más complejos.

No obstante, los propios autores subrayan que el objetivo del trabajo no fue exclusivamente la clasificación del pie plano, sino una caracterización más amplia del arco medial y su relación con factores como edad y diferencias de género. Aunque la ecuación AI–AHR representa una contribución valiosa, aún persisten limitaciones: la dificultad para diferenciar entre pie plano fisiológico y pie plano patológico en etapas tempranas de la infancia, la falta de análisis dinámicos durante la marcha y la concentración de la muestra en determinadas regiones. A esto se suma el elevado costo del escáner UPod, que limita su adopción en entornos clínicos generales y restringe su aplicación a contextos de investigación. A pesar de estas barreras, el estudio demuestra que la captura tridimensional y el uso de índices estandarizados

constituyen un camino sólido hacia diagnósticos más objetivos y reproducibles, al mismo tiempo que enriquecen la comprensión del desarrollo del arco plantar en población pediátrica.

## 2.4. Limitaciones de los enfoques revisados

Los trabajos revisados presentan limitaciones que condicionan su aplicabilidad clínica y la solidez de sus resultados. En el caso de [27], aunque el sistema de escaneo digital demostró ser preciso, depende de una calibración mecánica constante, lo que dificulta su implementación rutinaria fuera de entornos controlados. Esta dependencia de ajustes técnicos reduce su practicidad en contextos de uso cotidiano.

Por su parte, [28] desarrolló un podoscopio digital de bajo costo que mostró altos niveles de precisión diagnóstica; sin embargo, su desempeño es sensible a las condiciones de iluminación en el entorno de captura. Esto implica que, en escenarios no controlados, los resultados pueden perder confiabilidad. De manera similar, [30] implementó algoritmos de clasificación basados en datos de presión plantar, pero la base de datos utilizada fue limitada, lo que restringe la generalización de los modelos a poblaciones más diversas.

## 2.5. Brecha encontrada

La propuesta de este trabajo supera limitaciones asociadas a la calibración mecánica y a la dependencia exclusiva de huellas 2D al incorporar datos tridimensionales. Al generar representaciones volumétricas del pie, se obtiene una medida más robusta y reproducible, lo que plantea una alternativa objetiva que combina accesibilidad con rigor técnico y amplía el potencial de aplicación clínica. No obstante, debido a que el procesamiento contempla una segmentación por color de piel, el sistema mantiene cierta sensibilidad a la iluminación ambiental. Este aspecto,

aunque no invalida el método, constituye un punto de mejora futura, para lo cual se sugiere reemplazar el criterio cromático por técnicas de segmentación basadas en aprendizaje profundo que reduzcan el efecto de la variabilidad lumínica.

---

---

## CAPÍTULO 3

---

# Marco teórico

### Contenido

---

<b>3.1. Introducción a la visión por computadora con profundidad</b>	<b>26</b>
<b>3.2. Principios físicos de la obtención de imágenes de profundidad</b>	<b>27</b>
3.2.1. Métodos comunes para obtener profundidad	28
3.2.1.1. Luz estructurada vs luz codificada	29
3.2.1.2. Profundidad estéreo	30
3.2.1.3. Tiempo de vuelo	30
<b>3.3. Funcionamiento de la Cámara Intel RealSense D435i</b>	<b>31</b>
3.3.1. Arquitectura del sistema estéreo:	31
3.3.2. Emisión infrarroja y seguridad ocular	32
<b>3.4. Matemática detrás de la visión estéreo</b>	<b>33</b>
3.4.1. Matriz de profundidad y triangulación estereoscópica	33
<b>3.5. Transformación de Datos a nube de puntos</b>	<b>35</b>
3.5.1. Formato de archivo: .ply	35
<b>3.6. Componentes y parámetros de un sistema de visión</b>	<b>38</b>
3.6.1. Profundidad de campo (depth of field, DoF)	38
3.6.2. Campo de visión (field of view, FOV)	39
3.6.3. Resolución	40
3.6.4. Distancia mínima y máxima de trabajo	43
<b>3.7. Software Intel RealSense (Software Development Kit, SDK)</b>	<b>43</b>
<b>3.8. Segmentación basada en color</b>	<b>44</b>
3.8.1. Fundamentos de los espacios de color	45

3.8.2. Espacio de color HSL y su uso . . . . .	46
<b>3.9. Redes neuronales y aprendizaje profundo con aplicación a visión por computadora . . . . .</b>	<b>47</b>
3.9.1. Fundamentos de las Redes Neuronales Artificiales . . . . .	47
3.9.2. Redes neuronales convolucionales (convolutional neural network, CNN) . . . . .	50
3.9.3. Aprendizaje por transferencia . . . . .	52
3.9.4. Arquitecturas comunes aplicadas a imágenes de profundidad	53

---

### 3.1. Introducción a la visión por computadora con profundidad

La visión por computadora está en todos lados: desde las cámaras inteligentes que tienen los coches o bien desde cualquier objeto que les permita “ver”, como son las cámaras. A partir de este contexto las cámaras son los ojos y el cerebro, la computadora. Si bien la visión por computadora extrae la información de la escena y utiliza esos datos para analizarlos con el fin de llevar a cabo una tarea, esta puede ir desde el reconocimiento de objetos hasta tareas más complejas, partiendo de modelos previamente entrenados con imágenes convencionales, es decir, RGB. Sin embargo, existen otro tipo de imágenes con las cuales se puede hacer una mejor detección de objetos: estas son llamadas imágenes RGB-D, ya que contienen cuatro canales: 3 para el color y uno extra para la profundidad.

La investigación con respecto al uso de estas imágenes de profundidad abarca desde sensores CMOS, como el presentado en [32]. Dicho sensor es capaz de generar imágenes de profundidad a partir de pulsos láser en el espectro infrarrojo cercano, mediante detección subfónica, en conjunto con la técnica de tiempo de vuelo (time of flight, ToF), la cual se explicará en su sección correspondiente.

Por otra parte, también existen estudios que utilizan sensores RGB-D dentro de la rama de la agricultura. En uno de ellos, se evalúan diferentes sensores RGB-D y métodos de procesamiento para localizar frutas [33]. Otra aplicación de las imágenes de profundidad es la detección de fatiga en conductores, combinando esta tecnología con técnicas de aprendizaje profundo [34]. Como se puede ver, la visión por computadora basada en imágenes de profundidad ha llegado para quedarse. Si bien parte del desarrollo se enfoca en crear sensores capaces de generar mapas de profundidad más precisos, estas imágenes tienen múltiples aplicaciones prácticas en diversos campos. Por ello, en este capítulo se abordarán temas claves para entender el funcionamiento de las imágenes de profundidad: desde sus principios físicos, pasando por la descripción de la cámara utilizada en este proyecto, hasta la librería que permite procesar los datos capturados por dicho dispositivo.

## **3.2. Principios físicos de la obtención de imágenes de profundidad**

Con respecto a la literatura existente, las imágenes de profundidad (depth image) son abordadas bajo distintos términos, como mapas de profundidad (depth maps), mapa de rayos (rays map), perfiles de superficie (surface profiles) tal como se menciona en [35]. En dicho trabajo destacan que una imagen de profundidad se caracteriza por estar representada en escalas de grises, donde cada píxel representa una distancia desde el sensor de la cámara al objeto en la escena.

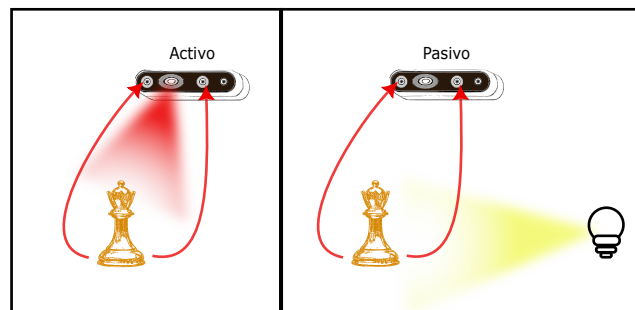
Existen dos técnicas que pertenecen a la familia de reconstrucción 3D, estas se basan en la intensidad de la imagen y el rango, conocidas como image-based y range-based. La reconstrucción basada en el rango mide la distancia entre el sensor y los puntos físicos del objeto en la escena, por otra parte la basada en la imagen utiliza la triangulación. Ambas técnicas son buenas, se puede observar en la siguiente tabla 3.1 algunas de sus diferencias [36], [37].

Característica	Imágenes de Intensidad	Imágenes de Rango
Velocidad de adquisición	Rápida adquisición	Lenta adquisición
Resolución espacial	Alta resolución	Resolución limitada
Procesamiento requerido	Requieren más procesamiento	Menor procesamiento

Tabla 3.1: Comparación entre imágenes de intensidad y de rango.

### 3.2.1. Métodos comunes para obtener profundidad

En [35] se explica que los sensores ópticos utilizados para capturar imágenes de profundidad suelen clasificarse en dos tipos: activos y pasivos. Esta distinción depende de si el sensor emite su propia fuente de iluminación (como un emisor infrarrojo) o si, por el contrario, se basa únicamente en la luz del entorno para calcular la profundidad, (ver Figura 3.1).



**Figura 3.1:** Clasificación de sensores con cámara de profundidad: activo (izquierda) y pasivo (derecha).

Los principales métodos físicos para obtener imágenes de profundidad incluyen [38]:

### 3.2.1.1. Luz estructurada vs luz codificada

La página oficial de Intel se presenta como una fuente útil, ya que ofrece desde ejemplos de aplicación hasta explicaciones claras sobre los principios básicos de la medición de profundidad. Además, describe las similitudes entre distintas tecnologías de captura 3D, así como las diferencias clave entre ellas. En la siguiente tabla 3.2 se resumen algunos de estos aspectos [39].

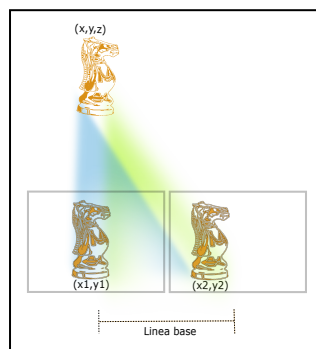
Tabla 3.2: Comparación entre luz estructurada y luz codificada.

Luz estructurada	Luz codificada
Se proyectan patrones geométricos conocidos (como franjas, rejillas o puntos) sobre el objeto.	Se proyectan secuencias de patrones especialmente diseñados (como códigos Gray, códigos de barras, ) para identificar cada punto de forma única, sin que exista confusión.
Analizar cómo se distorsiona el patrón al proyectarse sobre la superficie para estimar profundidad.	Se interpreta la información contenida en el patrón proyectado para identificar, con precisión, la ubicación espacial de cada punto en la superficie del objeto [40].

Como se ha visto, tanto la luz estructurada como la codificada permiten estimar la profundidad mediante la proyección de patrones específicos sobre una superficie. No obstante, existen otros enfoques que se basan en la comparación de imágenes capturadas desde dos puntos de vista distintos, cuya operación se detalla a continuación.

### 3.2.1.2. Profundidad estéreo

La luz infrarroja se utiliza comúnmente en dispositivos que emplean profundidad estéreo. Esta se calcula a partir de dos sensores ubicados a una distancia conocida como línea base, la cual corresponde a la separación entre los centros ópticos de ambos sensores. Gracias a ello, el dispositivo hace la comparación entre ambas imágenes de tal forma que así proporciona la profundidad (ver Figura 3.2) [39].

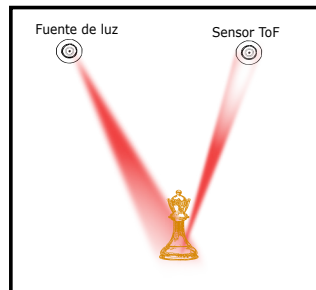


**Figura 3.2:** *Profundidad estéreo.*

Sin embargo, existen otras tecnologías que permiten medir la profundidad de forma diferente. Una de ellas es el método conocido como tiempo de vuelo, el cual será abordado a continuación.

### 3.2.1.3. Tiempo de vuelo

Las cámaras que utilizan esta técnica basan su cálculo en función del tiempo que tarda la luz en viajar desde su fuente hasta el objeto (ver Figura 3.3). Es importante mencionar que tanto la fuente como el sensor de ToF están integrados en el mismo dispositivo [38], [39].



**Figura 3.3:** *Técnica ToF.*

Como se ha explicado, las tecnologías activas como la luz estructurada, la luz codificada y la visión estéreo asistida por infrarrojo permiten estimar profundidad a partir de distintas estrategias.

En este trabajo se ha optado por no profundizar en los sensores pasivos (los cuales dependen únicamente de la iluminación ambiental) [37], ya que la cámara empleada, la Intel RealSense D435i, no pertenece a esta categoría.

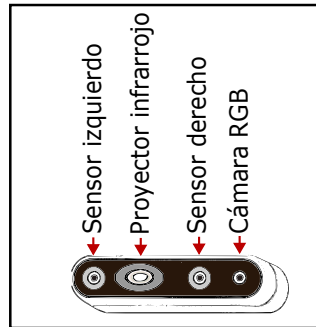
A continuación, se describe el funcionamiento específico de esta cámara.

### 3.3. Funcionamiento de la Cámara Intel RealSense D435i

#### 3.3.1. Arquitectura del sistema estéreo:

Con base en las tablas 3-10 y 3-13 del documento técnico, se describe el funcionamiento del módulo de profundidad de la cámara Intel RealSense D415i (ver Figura 3.4):

Este módulo incorpora dos sensores de imagen infrarrojos con un arreglo de filtro de color monocromático (Color Filter Array, CFA), ubicados en los extremos



**Figura 3.4:** *Componentes de la cámara.*

del dispositivo a una distancia de 50 mm. Entre ambos se encuentra un emisor de luz infrarroja encargado de proyectar un patrón estructurado sobre la escena.

De acuerdo con la tabla 2-1, este módulo de profundidad, en el modelo especificado, no integra un sensor de color.

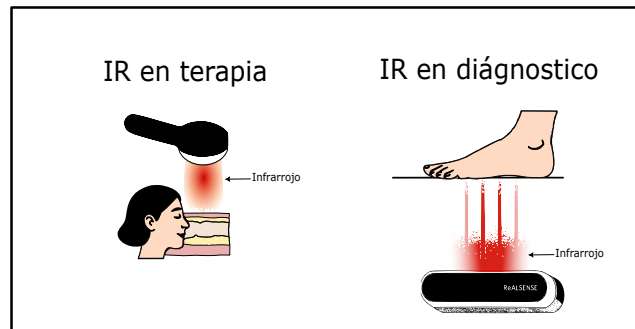
No obstante, la cámara incluye un sensor RGB independiente, situado junto al emisor de luz. Este sensor permite la captura de imágenes a color, lo cual resulta útil para complementar la información de profundidad.

### 3.3.2. Emisión infrarroja y seguridad ocular

La cámara Intel RealSense D435i emplea un proyector infrarrojo (IR) que emite un patrón de luz estático en el espectro IR cercano (850 nm). A diferencia de otras aplicaciones del infrarrojo cercano, como el fotorrejuvenecimiento [41], este dispositivo opera con una baja potencia, clasificada como Clase 1, lo que garantiza su seguridad incluso ante exposiciones prolongadas para el ojo humano y evita la penetración en tejidos profundos.

Este aspecto resulta particularmente relevante en el análisis plantar, ya que asegura que la proyección de luz infrarroja no representa un riesgo para la salud, ni genera efectos térmicos o de penetración en el tejido. Esto contrasta con sistemas

médicos específicamente diseñados con fines terapéuticos, en los cuales la potencia del IR es suficiente para interactuar con capas profundas de la piel (ver Figura 3.5).



**Figura 3.5:** Aplicación de la luz infrarroja en la piel.

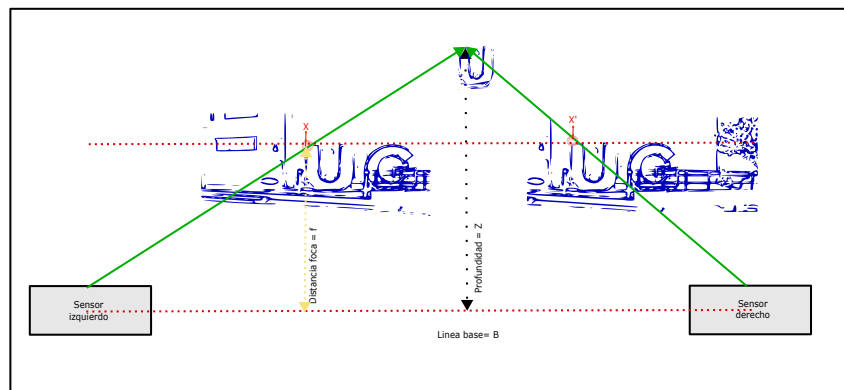
En la siguiente sección se abordará la matemática que sustenta la visión estéreo, profundizando en las ecuaciones que permiten estimar la posición espacial de cada punto capturado.

### 3.4. Matemática detrás de la visión estéreo

#### 3.4.1. Matriz de profundidad y triangulación estereoscópica

Una imagen constituye una representación bidimensional de una escena tridimensional de la cual se pueden extraer algunas características, por ejemplo, conociendo las dimensiones de un objeto es posible estimar la dimensión de otros objetos. Sin embargo, en esta transformación de tres a dos dimensiones se pierde la información de la profundidad, esto es, la distancia entre el observador y el objeto. Para recuperar de manera aproximada esta información es necesaria una segunda toma de la misma escena en una localización distinta, es decir, un segundo observador. Este concepto marca el inicio de la visión estéreo en visión por computadora.

La visión estéreo es una actividad común para el ser humano, esto debido a que el ser humano tiene dos ojos funcionales con los que aprecia el mundo desde dos perspectivas ligeramente distintas, con esto, el cerebro se encarga de interpretar las señales y dar un sentido de profundidad al escenario en el que se encuentra, por lo tanto, los objetos más cercanos se perciben con un desplazamiento más rápido que los objetos lejanos. Un sistema de visión estéreo con dos cámaras (izquierda y derecha) realiza una tarea similar: a partir de la distancia entre ambas cámaras, llamada línea base ( $B$ ), la cual generalmente es conocida, se calcula la profundidad ( $Z$ ). El mayor reto consiste en encontrar la similitud entre píxeles de la imagen izquierda y derecha, a esto se le nombra problema de correspondencia. De manera que resolverlo entrega un mapa de disparidad [42], el cual indica la diferencia entre las distancias de píxeles semejantes  $x$  y  $x'$ . Se tienen dos triángulos semejantes (ver Figura 3.6), uno de base  $B$  y altura  $Z$ , mientras el segundo está formado por la base  $x - x'$  y la altura  $Z - f$ . Gracias a la semejanza de estos triángulos, es posible escribir la siguiente relación [43]:



**Figura 3.6:** Representación visual del problema de correspondencia.

$$\frac{Z}{B} = \frac{Z - f}{B - (x - x')} \quad (3.1)$$

con la cual, despejando la variable  $Z$ , se llega a la fórmula para calcular la profundidad del objeto:

$$Z = \frac{Bf}{x - x'} \quad (3.2)$$

De la ecuación (3.2) se observa que la disparidad entre píxeles  $d \equiv x - x'$  es inversamente proporcional a la profundidad  $Z$ .

A partir de estas estimaciones, es posible representar la información de profundidad mediante nubes de puntos, lo cual será abordado en la sección siguiente.

## 3.5. Transformación de Datos a nube de puntos

### 3.5.1. Formato de archivo: .ply

En el proceso de transformación de imágenes de profundidad a nubes de puntos en tres dimensiones, se parte del modelo básico de formación de imagen de la cámara pinhole, el cual se describe ampliamente en [44].

En dicho modelo, la relación entre un punto en el espacio 3D y su proyección sobre la imagen está dada por la siguiente expresión en coordenadas homogéneas:

$$s \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \mathbf{K} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \quad (3.3)$$

donde  $\mathbf{K}$  corresponde a la matriz de calibración intrínseca de la cámara (ver ecuación (3.4)):

- $f_x, f_y$ : distancia focal expresada en píxeles (focal length in pixels).
- $c_x, c_y$ : coordenadas del centro óptico o punto principal (principal point).

$$\mathbf{K} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \quad (3.4)$$

y  $(u, v)$  representan las coordenadas de píxel en la imagen, mientras que  $(X, Y, Z)$  son las coordenadas del punto en el espacio 3D respecto al sistema de la cámara.

Para obtener la posición espacial  $(X, Y, Z)$  a partir de la imagen de profundidad, es necesario realizar la proyección inversa. La expresión resultante se puede escribir como:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = Z \cdot \mathbf{K}^{-1} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \quad (3.5)$$

Donde la matriz inversa  $\mathbf{K}^{-1}$  es:

$$\mathbf{K}^{-1} = \begin{bmatrix} \frac{1}{f_x} & 0 & -\frac{c_x}{f_x} \\ 0 & \frac{1}{f_y} & -\frac{c_y}{f_y} \\ 0 & 0 & 1 \end{bmatrix} \quad (3.6)$$

Se multiplica  $\mathbf{K}^{-1}$  por el vector  $[u, v, 1]^T$ :

$$\mathbf{K}^{-1} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{1}{f_x}u + 0 \cdot v + \left(-\frac{c_x}{f_x}\right) \cdot 1 \\ 0 \cdot u + \frac{1}{f_y}v + \left(-\frac{c_y}{f_y}\right) \cdot 1 \\ 0 \cdot u + 0 \cdot v + 1 \cdot 1 \end{bmatrix} = \begin{bmatrix} \frac{u-c_x}{f_x} \\ \frac{v-c_y}{f_y} \\ 1 \end{bmatrix} \quad (3.7)$$

Para escalar al mundo 3D, se multiplica por la profundidad conocida  $Z$ :

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = Z \cdot \begin{bmatrix} \frac{u-c_x}{f_x} \\ \frac{v-c_y}{f_y} \\ 1 \end{bmatrix} = \begin{bmatrix} Z \cdot \frac{u-c_x}{f_x} \\ Z \cdot \frac{v-c_y}{f_y} \\ Z \end{bmatrix} \quad (3.8)$$

Obteniendo así las coordenadas 3D  $(X, Y, Z)$ .

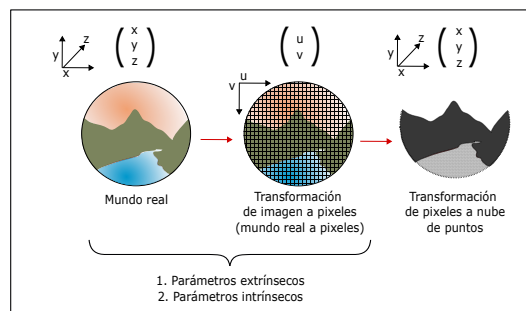
Este procedimiento es el que se implementa de forma práctica en diversos sistemas de visión por computadora. En particular, en [45] se ilustra de manera clara cómo realizar esta transformación en el contexto de cámaras Intel RealSense, partiendo de la imagen de profundidad y de los parámetros intrínsecos de la cámara. Las expresiones prácticas que se emplean se obtuvieron en la ecuación (3.8) y son:

$$X = Z \cdot \frac{(u - c_x)}{f_x} \quad (3.9)$$

$$Y = Z \cdot \frac{(v - c_y)}{f_y} \quad (3.10)$$

$$Z = Z \quad (3.11)$$

De este modo, es posible recuperar las coordenadas tridimensionales para cada píxel  $(u, v)$  de la imagen de profundidad, asociándoles además el valor de color correspondiente de la imagen RGB (ver Figura 3.7).



**Figura 3.7:** *Imagen real a nube de puntos.*

Una vez obtenida la nube de puntos, es común almacenarla en un formato estructurado que permita su posterior procesamiento o visualización.

Para comprender de manera integral el proceso de adquisición y representación de estos datos, es importante conocer también los componentes y parámetros que conforman un sistema de visión. Estos aspectos serán abordados en la sección siguiente.

## 3.6. Componentes y parámetros de un sistema de visión

### 3.6.1. Profundidad de campo (depth of field, DoF)

El concepto de profundidad de campo es sumamente relevante en la captura de imágenes, ya que determina cuán enfocada y nítida aparecerá la imagen resultante. En términos simples, la profundidad de campo corresponde al rango de distancias dentro de la escena en el cual los objetos aparecen enfocados de manera aceptable en la imagen.

Este efecto puede observarse claramente en la Figura 3.8a: en la imagen de la izquierda, únicamente la piedra con el logotipo de la universidad se encuentra enfocada, mientras que en la imagen de la derecha tanto la piedra como el árbol se observan enfocados. A la distancia entre los objetos que se encuentran dentro de este rango de enfoque aceptable se le denomina profundidad de campo [44].

Es importante destacar que la profundidad de campo no es un valor fijo, sino que puede variar en función de varios parámetros del sistema óptico y de la configuración de la toma. Los tres factores principales que influyen en la profundidad de campo son los siguientes:

- **Apertura:** La apertura del lente define el tamaño de la abertura a través de la cual pasa la luz hacia el sensor. Una apertura muy pequeña (similar al tamaño de un alfiler) incrementa la profundidad de campo, permitiendo que tanto el objeto principal como los elementos en el fondo se encuentren enfocados. Por el contrario, una apertura grande reducirá la profundidad de campo, haciendo que sólo el objeto de interés aparezca nítido mientras que el fondo se desenfoca.
- **Distancia al objeto:** La distancia entre la cámara y el objeto también afecta la profundidad de campo. Cuanto más cercana esté la cámara al objeto, menor será la profundidad de campo; en cambio, al aumentar la distancia entre la cámara y el objeto, la profundidad de campo se incrementa.
- **Distancia focal del lente:** La distancia focal es el parámetro que define el nivel de ampliación de la imagen. A mayor distancia focal, la profundidad de campo disminuye, produciendo un efecto de desenfoco en los planos alejados del objeto enfocado. Por el contrario, una menor distancia focal incrementa la profundidad de campo.

### 3.6.2. Campo de visión (field of view, FOV)

El campo de visión es una pieza clave para diseñadores de videojuegos, cineastas y para aquellas personas que trabajen con óptica. Básicamente, este concepto se describe a partir de la mayor área que logra capturar el sensor de la cámara mediante la apertura piramidal de la lente y el ángulo que subtiende, aquí es importante mencionar que los parámetros de mayor impacto en el FOV es el tamaño del sensor y la distancia focal, ya que estos están estrechamente relacionados y con base en ellos se determina si la lente tiene un campo de visión amplio o reducido; esto se ve claramente en la siguiente ecuación [44]:

$$FOV = 2 \tan^{-1} \left( \frac{H}{2f} \right) \quad (3.12)$$

donde  $H$  es la distancia medida en diagonal entre las esquinas del sensor y  $f$  es la distancia focal.



(a) *Profundidad de campo.*



(b) *Campo de visión.*

**Figura 3.8:** *Diferencias visuales para la comprensión de los términos profundidad de campo y campo de visión.*

### 3.6.3. Resolución

Para desarrollar el concepto de resolución, es primordial mencionar el proceso de conversión de luz a píxel. Este proceso consta de 7 etapas.

- Captura de luz (fotones): la luz de la escena ingresa a la cámara y llega al sensor (chip fotosensible). Cada partícula de luz (fotón) incide en pequeñas celdas llamadas fotodiodos (encargados de detectar la luz).
- Conversión a señal eléctrica (electrones): los fotones son convertidos en electrones (señal eléctrica) con una eficiencia llamada eficiencia cuántica (porcentaje de fotones que generan electrones). No toda el área del sensor es sensible a la luz debido a componentes internos (factor de relleno  $<50\%$  en sensores CMOS).
- Acumulación de carga (tiempo de exposición): los electrones se almacenan en un “depósito” (cubeta de carga) durante el tiempo de exposición (duración en que el sensor captura luz). Si hay demasiada luz, el depósito puede saturarse (desbordamiento).
- Amplificación y ruido: la señal eléctrica se amplifica para ser medida. En condiciones de poca luz, esto aumenta el ruido (granulado en la imagen), causado por electrones no generados por luz (corriente oscura).
- Digitalización: un conversor analógico-digital (analog-to-digital converter, ADC) transforma la señal eléctrica en un valor digital (número que representa el brillo del píxel). Algunos píxeles se usan como referencia para corregir errores (no uniformidad entre píxeles).
- Lectura y formación de la imagen: los valores digitales se leen en orden (de izquierda a derecha y arriba a abajo). Movimientos bruscos pueden causar desenfoque, por lo que se usan tiempos de exposición cortos.
- Imagen final (RGB o profundidad): los valores digitales se organizan para formar la imagen (color o mapa de profundidad), lista para procesarse en la computadora.

En sistemas de visión por computadora, la resolución está determinada no solo por el proceso de conversión de luz a píxeles, sino también por las características propias de los sensores utilizados. Una de sus ventajas de la cámara Intel RealSense D435i, es que se integran sensores infrarrojos y RGB. La Tabla 3.3 resume las

principales características técnicas de ambos sensores, las cuales son relevantes para entender las capacidades del sistema.

Tabla 3.3: Características de sensores RGB e Imagen OmniVision OV9282 en Intel RealSense D435i.

<b>RGB (1920×1080)</b>	<b>Imagen OmniVision OV9282 (1280×800)</b>
<p>Es la resolución tradicional para imágenes planas. El sensor RGB que contiene la cámara cuenta con las siguientes características [46]:</p> <ul style="list-style-type: none"> <li>▪ - Horizontal FoV: 69°</li> <li>▪ - Vertical FoV: 42°</li> </ul>	<p>Determina la calidad del cálculo de profundidad. Ambos sensores de imagen CMOS cuentan con las siguientes características [47]:</p> <ul style="list-style-type: none"> <li>▪ - Active pixels: 1280×800</li> <li>▪ - Focal length: 1.93 mm</li> <li>▪ - Horizontal FoV: 90°±1°</li> <li>▪ - Vertical FoV: 64°±1°</li> <li>▪ - Diagonal FoV: 98°±1°</li> </ul>

Los sensores OmniVision OV9282 cuentan con un tamaño de 1/4", parámetro que determina directamente las dimensiones en columnas y filas de la resolución del sensor, este tiene una resolución nativa de 1280x800, donde el tamaño del píxel individual es de 3 $\mu$ m.

Existe la resolución espacial, la cual se da a través de la correspondencia entre el objeto y el sensor de la imagen, utilizando unidades de *mm/pixel* y esto está estrechamente relacionado con el sensor y el campo de visión [48].

Matemáticamente se define como:

$$R_s = \frac{FOV}{R_c} \quad (3.13)$$

Este equilibrio entre precisión (mm/píxel) y cobertura (FoV) es crítico en aplicaciones como visión estereoscópica o inspección industrial, donde la elección del sensor depende de los requisitos específicos de detalle y alcance.

#### **3.6.4. Distancia mínima y máxima de trabajo**

La precisión de la profundidad implica un rango donde la medición es óptima antes de que el error domine, esto garantiza el mejor rendimiento del dispositivo. Según la hoja de especificaciones de Intel, la cámara D435i [46], opera dentro del rango que se extiende desde 0.2 m (MinZ) hasta 3 m. La distancia mínima (MinZ) no es un valor arbitrario, sino que deriva de limitaciones físicas en el cálculo de disparidad [49]. Esto implica que objetos más cercanos que el MinZ generan disparidades no detectables, haciendo imposible el cálculo de profundidad, por lo que fuera de este la precisión disminuye progresivamente.

Comprender los componentes que forman a un sistema de visión es indispensable para asegurar resultados óptimos, la siguiente sección abordará un conjunto de herramientas que proporciona Intel para procesar los datos de profundidad.

### **3.7. Software Intel RealSense (Software Development Kit, SDK)**

Para explotar al máximo las capacidades de la cámara D435i, Intel ofrece el RealSense SDK, un conjunto de herramientas de software que permiten configurar diferentes parámetros para integrarlos en aplicaciones personalizadas.

- **Adquisición de datos:**

Captura de streams de profundidad, RGB, infrarrojos y datos de la unidad de medición inercial (inertial measurement unit, IMU) en modelos como la D435i.

Control de parámetros en tiempo real (exposición, ganancia, modo de profundidad).

- Adquisición de datos:

Captura de streams de profundidad, RGB, infrarrojos y datos del IMU (en modelos como la D435i).

Control de parámetros en tiempo real (exposición, ganancia, modo de profundidad).

- Procesamiento avanzado:

Generación de nubes de puntos 3D.

Alineación de streams (ej: profundidad + RGB).

Filtrado de ruido y postprocesamiento de profundidad.

Compatibilidad multiplataforma:

Soporte para Windows, Linux (Ubuntu), Android y sistemas embebidos como Jetson.

APIs en C++, Python, C#, y wrappers para ROS, Unity y MATLAB.

### 3.8. Segmentación basada en color

La percepción del color constituye un mecanismo biológico esencial para la interpretación del entorno [50], por lo que se ha vuelto cada vez más común trasladar esta capacidad a sistemas artificiales mediante técnicas de visión por computadora.

A la agrupación de píxeles de acuerdo con propiedades únicas que describen el objeto a reconocer se le conoce como segmentación. La implementación de la segmentación basada en color adquiere relevancia en aplicaciones de clasificación automática, donde los píxeles proporcionan información de color, textura, forma, entre otras características. Por lo tanto, este tipo de métodos depende críticamen-

te del espacio de color seleccionado para el análisis. A continuación se abordarán conceptos clave.

### 3.8.1. Fundamentos de los espacios de color

Como fundamenta [51], un espacio de color constituye formalmente un sistema coordinado tridimensional que establece una correspondencia biunívoca, es decir, cada color perceptible corresponde a un único punto en el espacio y que cada punto válido representa un color realizable.

La implementación práctica exige los tipos de espacio de color mostrados en la Tabla 3.4, donde el espacio RGB es ampliamente utilizado en equipos de adquisición y visualización de imágenes, sin embargo, presenta una fuerte correlación entre sus canales, aumentando la complejidad de la segmentación. En cambio, el espacio HSL (Matiz, Saturación, Luminocidad/Luminancia - Hue, Saturation, Lightness/Luminance) muestra una separación clara entre los canales de matiz y saturación contra el canal de luminosidad, agregando una mayor robustez ante cambios de iluminación.

Tabla 3.4: Espacios de color comúnmente utilizados según la aplicación

Orientado a hardware	Orientado a percepción y procesamiento
<ul style="list-style-type: none"> <li>▪ RGB (Red, Green, Blue)</li> <li>▪ CMY (Cyan, Magenta, Yellow)</li> </ul>	<ul style="list-style-type: none"> <li>▪ HSV/HSL (Hue, Saturation, Lightness/Luminance)</li> <li>▪ YCbCr</li> </ul>

En [51] se resalta cómo los espacios perceptuales superan a RGB en aplicaciones de análisis visual. Mientras que la representación geométrica de RGB es un cubo, donde cada eje corresponde a un canal por color, con dominio normalizado entre 0 y 1. Los espacios perceptuales como HSV/HSL son representados por cilindros o conos dobles, separando matemáticamente los atributos de tono, saturación y luminosidad.

### 3.8.2. Espacio de color HSL y su uso

Anteriormente se mencionó el espacio HSL, pero también es conocido como HSI (Matiz, Saturación, Intensidad - Hue, Saturation, Intensity) [52]. Los intervalos de cada componente son:

- Tono (H): Rango  $[0, 2\pi]$ . Representa el color espectral puro en un dominio angular continuo. Suponga un color rojo, este estará en  $H = 0^\circ$ .
- Saturación (S): Rango normalizado  $[0, 1]$ . Cuantifica la pureza del color, es decir, un color rojo con  $S = 0.3$  se percibe como “deslavado”, mientras que con  $S = 0.9$  se percibe vibrante.
- Luminosidad (L): Rango discreto  $[0, 255]$  o normalizado. Cuantifica la iluminación del tono, por ejemplo, un valor de  $L = 0.2$  muestra un rojo oscuro, a diferencia de un  $L = 0.5$  que mostrará un rojo equilibrado.

En la literatura podemos encontrar ejemplos del uso del espacio HSL para la segmentación de núcleos de carcinoma en células transicionales [53], así mismo a partir de los canales S y L en [54] se basaron para clasificar las columnas de humo con el fin de detectar incendios forestales.

Para mayor detalle acerca de la conversión de RGB a HSL existen las siguientes referencias [51, 52].

## 3.9. Redes neuronales y aprendizaje profundo con aplicación a visión por computadora

### 3.9.1. Fundamentos de las Redes Neuronales Artificiales

En [55] describen que una red neuronal artificial a través de modelos matemáticos emula computacionalmente los mecanismos biológicos del cerebro, es decir, procesos de aprendizaje y toma de decisiones.

El origen de estas redes puede rastrearse hasta la regla de Hebb, uno de los primeros enfoques propuestos para modelar el aprendizaje sináptico. Aunque en la actualidad se emplean algoritmos más sofisticados, esta teoría sentó las bases conceptuales al proponer que la fortaleza de la conexión entre dos neuronas se incrementa cuando ambas se activan simultáneamente. En su formulación original, el estado de activación de una neurona se obtenía mediante el producto escalar entre los pesos sinápticos y las entradas, reforzando las conexiones con base en la frecuencia de uso.

A diferencia de los modelos computacionales, el cerebro humano opera de forma masivamente paralela. Cada una de las aproximadamente 100 mil millones de neuronas en el sistema nervioso mantiene una actividad constante, sin depender de ciclos o turnos de procesamiento como sucede en una CPU tradicional. Además, cada neurona está interconectada, en promedio, con unas 10,000 más a través de sinapsis, lo que permite una transmisión eficiente y simultánea de señales [55].

En contraste, una red neuronal artificial (RNA) debe actualizar el estado de sus neuronas de manera secuencial, procesándolas una a una a través de los recursos disponibles del procesador. Este comportamiento serial obliga a organizar las neuronas virtuales siguiendo una estructura definida, lo cual es necesario para optimizar el rendimiento y viabilidad computacional del modelo. Por ello, en la práctica, los algoritmos de aprendizaje profundo imponen restricciones específicas sobre la arqui-

itectura de la red, con el fin de facilitar su entrenamiento y escalabilidad en sistemas digitales.

Para calcular el estado de activación de una neurona en una red neuronal artificial, es necesario considerar las señales provenientes de las neuronas ubicadas en la capa anterior. Estas señales, conocidas como entradas, son ponderadas individualmente mediante pesos sinápticos y se combinan mediante un producto punto.

En el caso de una red totalmente conectada, cada neurona de una capa  $\ell$  recibe señales de todas las neuronas de la capa inmediatamente anterior, denotada como  $\ell - 1$ . Si esta capa tiene  $M_{\ell-1}$  neuronas, entonces cada neurona  $n_j$  de la capa  $\ell$  realiza la siguiente operación [56]:

$$u_j = \sum_{i=1}^{M_{\ell-1}} z_i w_{j,i} + w_{j,b} \quad (3.14)$$

donde  $z_i$  representa la salida de la neurona  $i$  en la capa anterior,  $w_{j,i}$  es el peso que conecta dicha neurona con la neurona  $j$  de la capa actual, y  $w_{j,b}$  es un término adicional correspondiente al sesgo o “bias”, asociado a la neurona  $j$ .

Este sesgo puede entenderse como una entrada constante de valor 1, que tiene su propio peso y permite ajustar el umbral de activación de la neurona. Por esta razón, cada neurona requiere  $M_{\ell-1} + 1$  pesos: uno para cada conexión con las neuronas anteriores, más uno correspondiente al bias.

En consecuencia, si la capa  $\ell$  contiene  $M_\ell$  neuronas, el total de pesos necesarios para conectar esta capa con la anterior es  $M_\ell \cdot (M_{\ell-1} + 1)$ . Esta estructura garantiza que cada neurona reciba toda la información relevante de la capa previa, más la flexibilidad que aporta el término de sesgo para ajustar la activación.

Una vez que se ha calculado la combinación ponderada de las entradas para cada neurona, el siguiente paso es aplicar una función que determine su nivel de

activación. Esta función es conocida como función de activación y es esencial para definir la salida final de la neurona.

El resultado de la suma ponderada, comúnmente representado por  $u_j$ , no se utiliza directamente como salida. En cambio, se transforma mediante una función no lineal  $\sigma$ , lo que produce la salida  $z_j$  de la neurona [55]:

$$z_j = \sigma(u_j) = \sigma \left( \sum_{i=1}^{M_{\ell-1}} z_i w_{j,i} + w_{j,b} \right) \quad (3.15)$$

Esta transformación es crucial ya que impide que la red neuronal funcione únicamente como un sistema lineal. Las funciones de activación permiten a la red capturar patrones complejos y modelar relaciones no lineales en los datos. Además, ayudan a controlar la magnitud de las salidas, especialmente cuando es necesario que estén en un rango acotado, como  $[-1, 1]$  o  $[0, 1]$ .

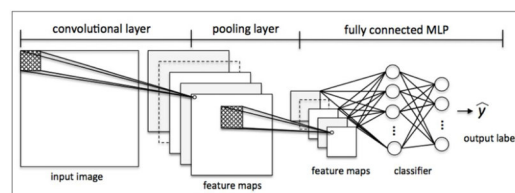
Existen diversas funciones de activación, pero algunas de las más utilizadas incluyen:

- **Tangente hiperbólica (tanh):** Produce valores en el intervalo  $[-1, 1]$  y está centrada en cero.
- **Sigmoide ( $\sigma$ ):** Devuelve valores entre 0 y 1, siendo históricamente una de las más empleadas en redes neuronales iniciales.
- **Unidad lineal rectificadora (rectified linear unit, ReLU):** Activa sólo cuando la entrada es positiva, lo que la hace computacionalmente eficiente y efectiva en redes profundas.

Cada función tiene características particulares que la hacen adecuada según el tipo de problema y arquitectura de la red. La elección correcta puede influir directamente en la capacidad de aprendizaje y generalización del modelo

### 3.9.2. Redes neuronales convolucionales (convolutional neural network, CNN)

A diferencia de las redes neuronales tradicionales en las que los datos de entrada, como las imágenes, se desenrollan y se convierten en vectores planos, perdiendo completamente su estructura espacial, las redes neuronales convolucionales trabajan de forma distinta: aprovechan esa estructura espacial original. En lugar de conectar cada píxel a todas las neuronas de la siguiente capa (como en una red completamente conectada), las CNN utilizan un enfoque más local y eficiente: pequeñas ventanas llamadas campos receptivos se deslizan sobre la imagen de entrada (ver Figura 3.9), lo que nos permite identificar patrones locales como bordes, esquinas o texturas simples [4].



**Figura 3.9:** Ilustración de red convolucional (tomada de [4]).

Imaginemos una imagen de entrada de  $128 \times 128$  píxeles con tres canales (RGB). En lugar de procesar directamente los casi 50 mil valores que la componen, aplicamos un pequeño filtro, también llamado kernel —por ejemplo, uno de tamaño  $3 \times 3$ — que se superpone parcialmente sobre la imagen y realiza una operación conocida como convolución. Este kernel puede tener valores diseñados (o aprendidos) para resaltar ciertas características, como detectar líneas horizontales, verticales, o zonas de contraste. Al mover este filtro por la imagen, se genera una nueva matriz de salida, conocida como mapa de características, que destaca las zonas donde se encuentra el patrón que el filtro está buscando.

Una ventaja clave de este enfoque es que ese mismo kernel se aplica a toda la imagen. Esto significa que si un borde o figura aparece en una esquina o en el

centro, el filtro tiene la misma capacidad de detectarlo. Esta reutilización de pesos no solo mejora la eficiencia computacional, sino que también reduce drásticamente la cantidad de parámetros que deben aprenderse, facilitando el entrenamiento de redes más profundas.

Después de esta primera capa de convolución, normalmente se incluye una capa de agrupamiento (pooling), como el max pooling. En lugar de buscar patrones, esta capa simplifica la información extraída: toma regiones del mapa de características y selecciona el valor más alto de cada una, reduciendo el tamaño del mapa sin perder lo más importante. Por ejemplo, si tras aplicar el kernel tenemos un mapa de  $126 \times 126$  (por no aplicar padding), el pooling puede reducirlo a  $63 \times 63$ . Esta compresión ayuda a evitar sobreajuste y acelera el procesamiento en las capas siguientes.

Aunque en este ejemplo solo usamos un único kernel sobre una imagen  $128 \times 128$  píxeles con tres canales (RGB), en redes reales se usan múltiples kernels en cada capa, generando varios mapas de características simultáneamente. Esto permite que las redes aprendan distintos patrones a la vez y mejore la toma de decisiones capa tras capa. Así, conforme se apilan más capas de convolución y agrupamiento, las redes comienzan a reconocer estructuras más complejas: desde líneas simples hasta formas completas y objetos enteros.

Finalmente, toda esta información resumida pasa a una o varias capas totalmente conectadas, que ya no se enfocan en extraer características, sino en clasificar. En este punto, el sistema ha convertido la imagen original en una representación abstracta compuesta por valores clave, y estos valores se usan para decidir a qué clase pertenece la imagen: un número, un objeto, una letra, etc.

Las redes convolucionales son una solución elegante y poderosa para trabajar con imágenes, ya que aprovechan la estructura espacial de los datos, reducen los parámetros y permiten una detección robusta de patrones en cualquier parte de la imagen.

### 3.9.3. Aprendizaje por transferencia

El aprendizaje por transferencia, conocido como transfer learning, es una técnica que ha cobrado mucha relevancia en los últimos años, sobre todo en tareas donde no se cuenta con una gran cantidad de datos. La idea principal es tomar un modelo ya entrenado en un dominio o tarea, y reutilizar lo que ese modelo “aprendió” para aplicarlo en otra tarea diferente, aunque relacionada [57].

En el documento [58], se hace un análisis bastante completo sobre esta estrategia. En la sección II, los autores explican que no todo transfer learning es igual. De hecho, lo clasifican en varios tipos según cómo se relacionan los datos y las tareas del modelo original con los del nuevo problema que se quiere resolver. Por ejemplo, hablan del homogeneous transfer learning, que es cuando los datos del modelo original y los del nuevo tienen una representación similar (como imágenes con el mismo tamaño y tipo de características). También mencionan el heterogeneous transfer learning, donde esa representación ya no es la misma. Además, hacen distinciones entre aprendizaje inductivo y transductivo, dependiendo de si la tarea cambia o no entre un dominio y otro. Esta parte es útil para entender en qué casos sí conviene aplicar esta técnica y en cuáles hay que tener cuidado.

Luego, en la sección IV, el artículo entra en un tema muy importante: cuándo es buena idea usar transfer learning y cuándo puede salir contraproducente. En general, los autores dicen que esta técnica es muy útil cuando no se tiene un gran conjunto de datos para entrenar desde cero, pero sí se cuenta con un modelo que fue entrenado en una tarea parecida. En esos casos, reutilizar parte del modelo puede ayudar a ahorrar tiempo, mejorar los resultados y evitar problemas como el sobreajuste.

Eso sí, también advierten que no siempre es buena idea usar esta técnica a la ligera. Si el nuevo problema que queremos resolver no tiene nada que ver con el modelo original, lo más probable es que el resultado sea peor. Esto se conoce como negative transfer, y puede pasar cuando las diferencias entre los dos dominios son demasiado grandes. Por eso, antes de aplicar transfer learning, conviene analizar bien qué tan relacionados están los datos y las tareas.

En resumen, ese artículo ayuda a entender no solo los tipos de transfer learning que existen, sino también los riesgos y ventajas que tiene dependiendo del contexto. Para este proyecto, donde se busca entrenar modelos con pocos datos disponibles, estas ideas resultan especialmente útiles.

#### 3.9.4. Arquitecturas comunes aplicadas a imágenes de profundidad

En el artículo [59] analizado, se presenta un enfoque interesante para la clasificación de frutas utilizando imágenes de profundidad. Este tipo de imágenes tiene la ventaja de ofrecer información estructural del objeto, como su forma y distancia, sin depender tanto de la iluminación del entorno, lo cual resulta muy útil en escenarios donde las condiciones de luz pueden variar.

Para lograr una mejor precisión en el reconocimiento, los autores implementaron redes neuronales convolucionales, ya que estas redes han demostrado ser muy eficientes en el análisis de imágenes complejas. Además, optaron por aplicar transfer learning, es decir, reutilizar modelos ya entrenados con grandes bases de datos para adaptarlos a esta tarea específica. Esto no solo reduce el tiempo de entrenamiento, sino que también permite obtener buenos resultados con menos datos.

Dado que las bases de datos en este tipo de tareas suelen ser limitadas, también emplearon técnicas de aumento de datos, como rotaciones, traslaciones y espejeo de imágenes. Estas transformaciones ayudan a generar más ejemplos a partir del mismo conjunto original, lo que fortalece el entrenamiento del modelo y mejora su capacidad para generalizar.

Entre los modelos utilizados, destacan arquitecturas profundas como ResNet101 y GoogleNet. Estas redes cuentan con muchas capas que permiten capturar patrones más complejos en las imágenes, lo cual se refleja en una mejor precisión al clasificar.

En conjunto, el estudio demuestra que al combinar imágenes de profundidad con modelos potentes y estrategias como el aprendizaje por transferencia, se pue-

den obtener resultados sólidos en tareas de reconocimiento. Esto abre posibilidades interesantes en aplicaciones como la robótica, la agricultura automatizada y los sistemas de visión en tiempo real.

---

## CAPÍTULO 4

---

# Desarrollo

### Contenido

---

<b>4.1. Descripción del esquema experimental . . . . .</b>	<b>57</b>
4.1.1. Configuración del sistema de medición . . . . .	57
4.1.2. Adquisición de mediciones tras la implementación de la estructura metálica con superficie de cristal templado . . . . .	58
<b>4.2. Desarrollo del sistema de interfaz para adquisición de datos . . . . .</b>	<b>61</b>
<b>4.3. Población muestra . . . . .</b>	<b>64</b>
<b>4.4. Pruebas digitales de los métodos en la literatura . . . . .</b>	<b>65</b>
4.4.1. Índice de Staheli . . . . .	66
4.4.2. Índice de Chippaux-Smirak . . . . .	68
<b>4.5. Preprocesamiento de las imágenes . . . . .</b>	<b>70</b>
<b>4.6. Área bajo la curva . . . . .</b>	<b>73</b>
<b>4.7. Experimento 1: Clasificador Bayesiano con KDE sobre el Factor Standard . . . . .</b>	<b>77</b>
4.7.1. Desarrollo del método . . . . .	77
4.7.1.1. Planteamiento general . . . . .	77
4.7.1.2. Datos y particionado . . . . .	78
4.7.1.3. Modelado KDE por clase . . . . .	78
4.7.1.4. Regla bayesiana de decisión . . . . .	78
<b>4.8. Experimento 2: Truncado balanceado (13/13/13) y Bayes-KDE . . . . .</b>	<b>79</b>
4.8.1. Planteamiento y objetivo . . . . .	79
4.8.2. Datos y particionado . . . . .	80

---

4.8.3. Modelado KDE por clase . . . . .	80
4.8.4. Regla bayesiana . . . . .	81
<b>4.9. Experimento 3: KDE + Bayes con clases 0 y 1 . . . . .</b>	<b>81</b>
4.9.1. Planteamiento y objetivo . . . . .	81
4.9.2. Distribución de datos . . . . .	82
4.9.3. Estimación KDE por clase . . . . .	82
4.9.4. Regla bayesiana de decisión . . . . .	83
<b>4.10. Experimento 4: MLP con doble particionado y particio-</b>	
<b>nado simple . . . . .</b>	<b>83</b>
4.10.1. Planteamiento y objetivo . . . . .	83
4.10.2. Datos y particionado . . . . .	84
4.10.3. Preprocesamiento y aplanamiento de matrices . . . . .	84
4.10.4. Modelo MLP y parámetros de entrenamiento . . . . .	85
<b>4.11. Experimento 5: MLP sobre la matriz de profundidad de</b>	
<b>todo el pie . . . . .</b>	<b>86</b>
4.11.1. Planteamiento y objetivo . . . . .	86
4.11.2. Datos y particionado . . . . .	87
4.11.3. Modelo MLP y parámetros de entrenamiento . . . . .	87
<b>4.12. Experimento 6: Extracción de vectores de características</b>	<b>88</b>
4.12.1. Planteamiento y objetivo . . . . .	88
4.12.2. Datos y particionado . . . . .	89
4.12.3. Modelos y parámetros de entrenamiento . . . . .	89
<b>4.13. Experimento 7: Matriz de profundidad + vectores de</b>	
<b>características (pie derecho real) . . . . .</b>	<b>91</b>
4.13.1. Planteamiento y objetivo . . . . .	91
4.13.2. Datos y particionado . . . . .	92
4.13.3. Modelos y parámetros de entrenamiento . . . . .	93
<b>4.14. Experimento 8: vectores de características con balanceo</b>	
<b>mediante SMOTE . . . . .</b>	<b>94</b>
4.14.1. Planteamiento y objetivo . . . . .	94
4.14.2. Datos y particionado . . . . .	95
4.14.3. Modelos y parámetros de entrenamiento . . . . .	96

---

## 4.1. Descripción del esquema experimental

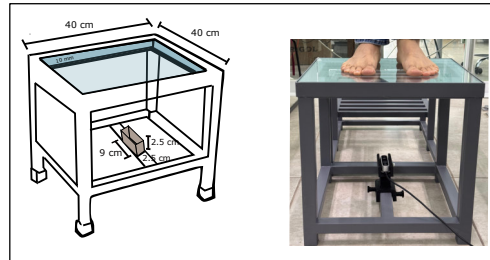
### 4.1.1. Configuración del sistema de medición

Se utilizó una cámara de profundidad para las mediciones iniciales. Se realizaron pruebas colocando la cámara paralela a un pie fabricado en impresora 3D, manteniendo una distancia de 39.4 cm entre el lente de la cámara y la superficie inferior del cristal. Se analizaron seis zonas (denominadas A-F) en la imagen RGB-D, observándose variaciones significativas.

Estos resultados confirmaron que el sistema era sensible a las diferencias de profundidad, por lo que se consideró viable avanzar hacia el diseño de un esquema experimental más controlado. A partir de ello, se procedió con el diseño y construcción de una estructura metálica que permitiría estandarizar las condiciones de trabajo. Las dimensiones de dicha estructura fueron de 40 cm de ancho por 40 cm de largo y 45 cm de altura. Así, las medidas del área útil (ancho y largo) fueron propuestas con base en las dimensiones típicas de un plantoscopio comercial; sin embargo, la altura se definió considerando el rango de captura mínimo de la cámara de profundidad.

Sobre esta estructura se instaló un cristal templado de 10 mm de espesor, con una capacidad de carga de hasta 150 kg, permitiendo que los sujetos se colocaran directamente sobre él para la adquisición de datos.

Además, se diseñó e imprimió un instrumento a la medida para posicionar la cámara utilizando una impresora 3D, con el fin de asegurar su correcta colocación y alineación. Esta base plástica midió 9 cm de largo por 2.5 cm de ancho y 2.5 cm de altura, y fue posicionada de forma centrada en la parte inferior de la estructura metálica (ver Figura 4.1).



**Figura 4.1:** *Fotografía de la estructura física y representación digital del diseño desarrollado para la adquisición de imágenes de profundidad*

#### 4.1.2. Adquisición de mediciones tras la implementación de la estructura metálica con superficie de cristal templado

El desarrollo del programa se llevó a cabo en lenguaje Python, específicamente en su versión 3.8.20, por ser compatible con la biblioteca `pyrealsense2`, necesaria para establecer comunicación con la cámara Intel® RealSense™ desde código. Para mantener una organización adecuada, se creó un entorno virtual dedicado exclusivamente a este proyecto dentro del entorno de desarrollo Spyder, el cual fue seleccionado por su capacidad para mostrar variables en tiempo real, lo cual facilitó la depuración y el análisis del comportamiento del sistema.

Antes de comenzar la programación, fue indispensable instalar el SDK oficial de Intel® RealSense™, que incluye el software RealSense Viewer. Este visor permitió verificar el reconocimiento y correcto funcionamiento de la cámara, asegurando que los sensores respondieran adecuadamente antes de integrarlos en el flujo de trabajo programado en Python.

Con la finalidad de comprobar el funcionamiento adecuado del sistema de adquisición de datos, se desarrolló un programa inicial tomando como base el código de ejemplo disponible en el repositorio oficial de Intel [60]. Dicho ejemplo permite alinear la imagen de profundidad con la imagen RGB obtenida por la cámara Intel RealSense, mostrando ambas simultáneamente en pantalla.

Si bien el código no fue estructurado inicialmente de forma modular, su lógica interna permite identificar bloques funcionales definidos, que abarcan desde la inicialización del sistema hasta la visualización y almacenamiento de los datos.

El flujo del programa puede recordarse con la nemotecnia CIACA, que resume las etapas clave del sistema:

- **Configuración:** Se crea el pipeline (`rs.pipeline()`), se definen los flujos de entrada (`rs.config()`), y se verifica que la cámara cuente con sensor RGB mediante `get_info()`.
- **Inicio:** Se habilitan los flujos de color y profundidad con `enable_stream()`, y se inicia la transmisión usando `pipeline.start()`.
- **Alineación:** Se define un límite de distancia (`clipping_distance`) y se sincronizan los cuadros con `rs.align()`.
- **Captura y visualización:** Se capturan y alinean los cuadros (`wait_for_frames()` y `align.process()`), se convierten a matrices (`np.asanyarray()`), y se visualizan combinando las imágenes RGB y de profundidad coloreada (`cv2.convertScaleAbs()`, `cv2.applyColorMap()` y `cv2.imshow()`).
- **Almacenamiento:** Al finalizar, se guardan las imágenes y los datos de profundidad utilizando `cv2.imwrite()` y `np.savetxt()`.

Este enfoque funcional permitió comprobar que el sistema era capaz de capturar datos de color y profundidad de forma precisa y alineada, y además proporcionó una base sólida para el desarrollo de una interfaz gráfica de usuario, que se presentará en la sección 3.2.

Una vez obtenidas las imágenes RGB y los archivos de profundidad en formato .csv, se diseñó un segundo programa con fines exclusivamente visuales, cuyo objetivo fue facilitar la inspección puntual de las mediciones. Este código permitió seleccionar manualmente píxeles específicos dentro de la imagen y visualizar, en tiempo real, el valor de profundidad correspondiente en milímetros.

El funcionamiento general de este módulo puede resumirse con la nemotecnia CVDL:

- Carga de archivos: se lee la imagen RGB y el archivo .csv que contiene la matriz de profundidad.
- Visualización interactiva: se muestra la imagen en una ventana, y el usuario puede mover el cursor o hacer clic en distintos puntos.
- Detección de eventos: al hacer clic, el programa guarda la posición del punto y muestra la profundidad correspondiente. Además, a cada punto se le asigna una letra (como A, B, C...), y esa letra junto con el valor de profundidad se muestran sobre la imagen.
- Lectura rápida de datos: los puntos en los cuales se ha hecho clic aparecen también en la parte superior izquierda de la pantalla con su respectivo valor en milímetros, lo que facilita comparar varias regiones de forma sencilla.

Este segundo bloque no se utilizó en el desarrollo de la interfaz principal, pero resultó útil para comprobar la correspondencia entre los datos numéricos de profundidad y la imagen capturada.

La Figura 4.2 muestra un ejemplo del resultado obtenido con este módulo, donde pueden observarse distintos puntos seleccionados sobre la planta del pie, cada uno acompañado de su valor de profundidad correspondiente expresado en milímetros.



**Figura 4.2:** Visualización de puntos seleccionados manualmente sobre la imagen RGB.

## 4.2. Desarrollo del sistema de interfaz para adquisición de datos

Para facilitar el proceso de captura de datos e imágenes, se desarrolló una interfaz gráfica utilizando la biblioteca Streamlit, combinada con pyrealsense2, pandas, OpenCV y NumPy. Antes de su implementación, fue necesario asegurarse de instalar dichas bibliotecas dentro del entorno virtual, ya que cada una cumple una función esencial en la ejecución: streamlit permite el diseño de la interfaz web a nivel local, pandas facilita el manejo de datos tabulares, opencv se utiliza para el procesamiento y guardado de imágenes, y pyrealsense2 permite la conexión con la cámara Intel® RealSense™.

La aplicación está organizada en dos formularios (ver Figura 4.3). El primer formulario se encarga únicamente de capturar datos generales de la persona: sexo, edad, estatura (en centímetros) y peso (en kilogramos, con posibilidad de introducir decimales). Estos campos no solicitan ningún dato identificable, lo que garantiza el anonimato del participante. El botón de Guardar en este formulario no guarda automáticamente los datos al primer clic, sino que requiere ser presionado dos veces

## 4.2. DESARROLLO DEL SISTEMA DE INTERFAZ PARA ADQUISICIÓN DE DATOS<sup>62</sup>

como una forma de confirmación, permitiendo que el usuario revise y corrija los valores ingresados antes de que se registren. Al completarse este paso, el sistema asigna un identificador único (ID) al sujeto, el cual se asocia internamente a todos los archivos generados.

The figure displays two screenshots of a web interface. The left screenshot, titled "Intel RealSense - Captura de Datos e Imágenes", shows a registration form with the following fields: "Sexo:" with radio buttons for "Masculino" (selected), "Femenino", and "Otro"; "Edad:" with a text input field containing "12"; "Estatura:" with a text input field containing "155"; and "Peso:" with a text input field containing "45". A "Guardar Datos" button is at the bottom. The right screenshot, titled "Imágenes", shows a live camera view of a person's feet. It includes a "Capturar y Guardar Imágenes" button, a "Listo" button, and a confirmation message at the bottom: "Datos guardados correctamente con ID: 5".

**Figura 4.3:** *Formulario 1 (izquierda) y Formulario 2 (derecha).*

Una vez registrado el ID, se activa el segundo formulario, que tiene como función principal realizar la captura de imágenes mediante la cámara Intel® RealSense™. En esta etapa se muestra una vista en vivo de la cámara, permitiendo al operador visualizar exactamente lo que se va a capturar. El formulario cuenta únicamente con dos botones:

Capturar y Guardar Imágenes: al presionarlo, se almacena de forma automática la imagen RGB (a una resolución máxima de 1920×1080), el mapa de color, el archivo .csv con los datos de profundidad y la nube de puntos correspondiente en

formato .ply. Todos los archivos generados se guardan en carpetas específicas y están asociados al mismo ID.

Listo: este botón permite regresar al formulario de captura de datos para registrar a un nuevo sujeto. Al igual que el botón de guardado en el primer formulario, requiere un segundo clic para confirmar la acción y reiniciar el flujo de captura.

Desde el punto de vista técnico, este segundo formulario reutiliza la lógica del código presentado en la sección 3.1, especialmente en lo relacionado con la alineación de imágenes y la extracción de datos de profundidad. La principal diferencia radica en que ahora el código está ordenado en funciones modulares, y se integró la generación de la nube de puntos con la función `rs.pointcloud()` de la biblioteca `pyrealsense2`. Para esta funcionalidad, se tomaron como referencia los ejemplos oficiales disponibles en la documentación de Intel<sup>®</sup>. A diferencia de dichos ejemplos, donde es posible modificar la densidad de la nube de puntos, en este caso se optó por trabajar con la máxima resolución posible para conservar el mayor nivel de detalle. Si bien el valor exacto puede variar según el dispositivo y la configuración, se utilizó un nivel de muestreo equivalente a la densidad más alta disponible [61].

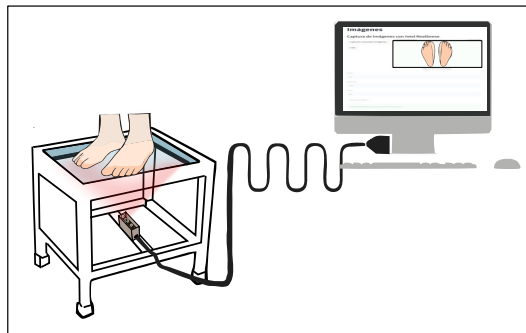
Además, en esta etapa no se aplicó el umbral de `clipping_distance`, ya que el objetivo era conservar los datos originales sin limitar la profundidad capturada. De esta forma, se obtiene un conjunto completo de información útil para análisis posteriores.

Para facilitar la comprensión de este flujo, se propone la nemotecnia DICAP, que resume sus cinco componentes principales:

- Datos: ingreso de sexo, edad, estatura y peso en un formulario anónimo.
- Inicialización de la cámara: configuración del pipeline y validación del sensor RGB.
- Captura en vivo: visualización en tiempo real de la imagen desde la cámara.

- Almacenamiento: guardado automático de todos los archivos asociados (imagen RGB, mapa de color, profundidad y nube de puntos).
- Punto de reinicio: retorno al formulario inicial para registrar un nuevo sujeto.

La Figura 4.4 muestra el funcionamiento completo del sistema utilizado para capturar las fotografías de la planta de los pies. Se observa una estructura metálica con superficie de vidrio, donde la persona coloca ambos pies. Debajo del vidrio está instalada la cámara Intel® RealSense™, la cual apunta hacia arriba para registrar tanto la imagen RGB como el mapa de profundidad.



**Figura 4.4:** *Configuración experimental para la captura de datos.*

La cámara se conecta por USB a una computadora que ejecuta una interfaz local hecha con Streamlit. En la pantalla se muestra esta interfaz, desde donde es posible ver en tiempo real la imagen capturada y guardar los datos con solo presionar un botón. Este flujo simplifica el proceso de captura sin perder calidad en los resultados.

### 4.3. Población muestra

La muestra estuvo formada por 100 personas voluntarias, 17 mujeres y 83 hombres. A cada participante se le asignó un ID numérico para mantener el anonimato,

y no se pidió ningún dato que pudiera usarse para identificarlos. Los únicos datos registrados fueron edad, estatura (en centímetros), peso (en kilogramos, incluyendo decimales) y sexo.

Todos los voluntarios tenían entre 18 y 21 años. Se les explicó con claridad que los datos recolectados serían utilizados para fines científicos y formarían parte de un dataset de libre acceso. Cada persona aceptó participar de forma consciente y firmó un documento de consentimiento antes de iniciar.

Durante la toma de datos se siguieron medidas básicas de higiene. Por ejemplo, se les pidió quitarse los zapatos para medir su estatura, y entre cada registro se desinfectó el equipo con Lysol para evitar contagios por hongos o bacterias.

Con esto se logró reunir un conjunto de datos ordenado, anónimo y útil para el análisis posterior.

#### 4.4. Pruebas digitales de los métodos en la literatura

Antes de implementar los índices morfométricos de huella plantar propuestos en la literatura, fue necesario establecer una conversión confiable entre píxeles y milímetros (px  $\rightarrow$  mm). Para ello, se colocó un objeto rectangular con dimensiones conocidas frente a la cámara, específicamente una tarjeta de 85.58 mm de largo y 54.12 mm de alto, medidas previamente con un calibrador Vernier digital. La tarjeta fue fotografiada a una resolución de 1920 $\times$ 1080 píxeles. Luego, se midió el ancho de la tarjeta en la imagen obtenida, resultando en 381 píxeles. De esta forma, se dedujo un factor de conversión de:

$$\text{Factor de conversión} = \frac{85.58 \text{ mm}}{381 \text{ px}} = 0.2246 \text{ mm/px} \quad (4.1)$$

Con este factor definido, se procedió a implementar el cálculo del Índice de Staheli, uno de los métodos más comunes en la literatura para la detección de pie plano, como ya se explicó en el Capítulo I. Su elección responde a su prevalencia en estudios clínicos y revisiones sistemáticas, junto con el índice de Chippaux.

#### 4.4.1. Índice de Staheli

Para facilitar la explicación del proceso, puede recordarse con la nemotecnia ABCD: *Aprecia Bien Cada Dato*, que resume las cuatro coordenadas necesarias para aplicar el método.

- A y B: puntos ubicados en la zona más angosta del mediopié, la cual, como se explicó anteriormente, está anatómicamente compuesta por el escafoide, las tres cuñas y el cuboide.
- C y D: puntos tomados desde la parte posterior del pie (retropié), específicamente los extremos más prominentes de los bordes interno y externo de la huella.

Una vez seleccionados estos puntos manualmente sobre la imagen, el código obtiene sus coordenadas exactas y calcula las distancias euclidianas AB y CD, primero en píxeles y posteriormente en milímetros, usando el factor de conversión anteriormente obtenido.

Posteriormente, se aplica la fórmula:

$$\text{Índice de Staheli} = \frac{\text{Ancho del mediopié (AB)}}{\text{Ancho del retropié (CD)}} \quad (4.2)$$

El valor resultante permite clasificar el tipo de pie de acuerdo con los rangos de la literatura:

Tabla 4.1: Clasificación del pie según el índice de Staheli

Rango del índice	Tipo de pie	Clase
$S < 0.44$	Pie cavo	2
$0.44 \leq S \leq 0.89$	Pie normal	0
$S > 0.89$	Pie plano	1

El resultado del índice se muestra en pantalla junto con los segmentos dibujados y las etiquetas A, B, C, D. En la esquina superior izquierda de la imagen se visualizan las coordenadas, las distancias calculadas y la clasificación obtenida. Un ejemplo de esta aplicación puede verse en la Figura 4.5, donde se ilustra el análisis correspondiente al ID 4 del pie derecho en imagen (equivalente al pie izquierdo físico). El índice resultó en un valor de 0.597, correspondiente a un pie de tipo normal, asignado a la clase 0.



**Figura 4.5:** Visualización de segmentos requeridos para el cálculo del índice Staheli [5].

Todos los datos, incluyendo coordenadas por punto, distancias en píxeles y milímetros, índice de Staheli y clase, se almacenan en un archivo .csv mediante un evento de teclado (tecla “g”). Esta acción también permite avanzar a la siguiente imagen del dataset para repetir el proceso. Aunque el procedimiento es manual, fue

diseñado para ser claro, ordenado y repetible. Se aplica exclusivamente sobre las huellas del pie izquierdo físico, que en pantalla aparece como pie derecho.

#### 4.4.2. Índice de Chippaux-Smirak

El Índice de Chippaux-Smirak, comúnmente abreviado como CSI por sus siglas en inglés, es una de las herramientas más empleadas para evaluar el tipo de pie y su grado de aplanamiento, a diferencia del Índice de Staheli que únicamente indica si hay o no planicidad. Este índice se calcula como la relación entre la anchura mínima de la región del arco del mediopié y la anchura máxima del antepié, todo expresado en porcentaje.

Para poder posicionar adecuadamente los puntos de medición, primero se trazaron dos líneas guía. La primera conecta la parte más prominente del antepié con la del retropié en el borde interno de la huella, mientras que la segunda conecta esas mismas partes pero por el borde externo. Estas líneas, aunque no son visibles en la imagen final, sirvieron como referencia para ubicar con precisión los cuatro puntos necesarios para el cálculo.

Una vez definidas estas referencias, se seleccionaron manualmente cuatro puntos, denominados A, B, C y D. Los puntos A y B se ubicaron sobre las partes más prominentes del antepié (interno y externo), formando el segmento AB. Por otro lado, los puntos C y D se colocaron sobre la región más angosta del arco del mediopié, formando así el segmento CD. Esto se hizo conforme a lo que dicta la literatura en cuanto a la correcta ubicación anatómica de estas referencias.

A nivel de código, se capturaron las coordenadas de estos cuatro puntos mediante eventos de clic izquierdo. Luego se calcularon las distancias euclidianas correspondientes a los segmentos AB y CD, primero en píxeles y después en milímetros, aplicando el mismo factor de conversión definido al inicio de la sección 3.4 para imágenes de resolución 1920×1080 (0.2246 mm/px).

De acuerdo con los segmentos mencionados y similar a la ecuación (2.1), el valor del índice se calcula con la siguiente fórmula:

$$\text{CSI} = \left( \frac{\text{CD}}{\text{AB}} \right) \times 100 \quad (4.3)$$

Una vez calculado el CSI, el sistema muestra en pantalla el valor del índice junto con las etiquetas A, B, C y D colocadas sobre la imagen, además de los segmentos y las distancias correspondientes. Toda esta información se despliega de manera visual para facilitar la interpretación, y en la esquina superior izquierda se imprimen también las coordenadas de cada punto, las distancias en píxeles y milímetros, así como la clasificación resultante.

Como ejemplo, en la Figura 4.6 se presenta el análisis correspondiente al ID 4 del pie derecho mostrado en pantalla, que en realidad corresponde al pie izquierdo físico. En ese caso, se obtuvo un valor de CSI de 35.2%, lo que indica un pie tipo normal, clasificado como clase 0 y grado 2 según los criterios de la literatura.



**Figura 4.6:** *Índice de Chippaux.*

El desglose completo de los valores —incluyendo las coordenadas de cada punto, las distancias AB y CD, el índice CSI y la clase asignada— se guarda automáticamente en un archivo .csv al presionar la tecla g. Esta acción también permite

avanzar de inmediato a la siguiente imagen del conjunto de datos para repetir el procedimiento. Aunque se trata de una metodología manual, fue pensada para ser clara, repetible y organizada.

Este proceso se aplicó exclusivamente sobre los pies izquierdos físicos (que visualmente aparecen como pies derechos en pantalla). Una vez terminada esta etapa, se realizó el mismo procedimiento sobre los pies derechos físicos y sus resultados se almacenaron en un archivo .csv diferente, respetando la misma estructura de columnas.

## 4.5. Preprocesamiento de las imágenes

Las imágenes obtenidas con la interfaz de adquisición se almacenaron en dos formatos por cada captura: un mapa de profundidad en formato CSV y la imagen RGB correspondiente en formato PNG. Antes de calcular las métricas morfológicas y entrenar los modelos de clasificación, fue necesario aplicar un preprocesamiento sistemático a este material, el cual se implementó en Python utilizando las bibliotecas NumPy, OpenCV, scikit-learn y pandas. A continuación se describen los pasos principales de este proceso.

En primer lugar, para cada par de archivos se cargó el mapa de profundidad y la imagen de color y se recortó una región de interés fija que contenía únicamente la plataforma de vidrio y las huellas plantares. Sobre el mapa de profundidad se aplicó un filtrado por distancia utilizando la escala de la cámara, de manera que sólo se conservaran los píxeles correspondientes a un intervalo específico alrededor del plano del cristal (aproximadamente entre 0.3 y 0.4 m). Los valores fuera de este rango, así como las lecturas nulas, se forzaron a cero para eliminar el fondo y posibles reflejos.

Posteriormente se realizó una primera segmentación de las huellas a partir del mapa de profundidad, generando una máscara binaria mediante umbrales adaptados al rango de distancias en el que se encontraba la planta del pie. Esta máscara permitió

extraer los contornos externos mediante las funciones de detección de contornos de OpenCV. De todos los contornos detectados se conservaron únicamente los dos de mayor área, que corresponden a los pies derecho e izquierdo sobre la plataforma.

Con el fin de refinar la segmentación, se aprovechó también la información de color. La imagen RGB se transformó al espacio de color HLS y se definieron rangos específicos de tono, luminosidad y saturación para distinguir la piel de la planta del pie del fondo y de la estructura de soporte. A partir de estas bandas se construyeron máscaras binarias que, tras operaciones morfológicas de apertura y cierre, permitieron suprimir ruido y recuperar de forma más limpia la región de interés. Combinando estas máscaras con los contornos de profundidad se obtuvo, para cada imagen, una máscara depurada de la huella plantar izquierda, que se utilizó en etapas posteriores para el cálculo del área y del volumen.

Sobre la huella izquierda se determinaron de manera automática tres puntos de referencia: un punto prominente en el antepié, el talón y un punto medio entre ambos. Para ello se recorrieron los píxeles blancos de la máscara, primero en la región superior para localizar la zona de antepié más adelantada y después en la mitad inferior de la imagen para identificar el punto más distal en el talón. A partir de la distancia vertical entre ambos puntos se calculó una coordenada intermedia que definió el punto medio de referencia. Esta geometría se utilizó más adelante para estimar una medida de altura relativa del arco.

Una vez segmentadas las huellas, se ajustó un plano de referencia sobre el mapa de profundidad recortado. El plano se estimó mediante regresión lineal del tipo  $z = ax + by + c$  utilizando únicamente los píxeles válidos de la región de soporte. La ecuación obtenida se empleó para corregir la inclinación del cristal: a cada valor de profundidad se le restó la contribución del plano ajustado, generando así un mapa de profundidad corregido donde el valor cero corresponde aproximadamente al nivel del cristal y los valores positivos representan la elevación de la planta del pie por encima de dicho plano.

Sobre este mapa corregido se llevó a cabo la estimación de la altura y el volumen bajo el arco. A partir del punto medio definido entre antepié y talón se recorrió horizontalmente la imagen hasta encontrar el último punto válido antes de que la máscara se anulara. La profundidad en ese punto se interpretó como una medida local de altura del arco en milímetros. La distancia vertical entre antepié y talón se convirtió de píxeles a milímetros mediante un factor de escala conocido de 0.2246 mm/píxel, y con ambos valores se calculó un índice de altura relativo (AHI). Por otro lado, la máscara binaria se empleó para contar los píxeles pertenecientes a la huella, estimando el área proyectada en  $\text{mm}^2$ , mientras que el volumen bajo la curva se obtuvo como la suma de las profundidades relativas (restadas respecto a la profundidad mínima) multiplicadas por el área de cada píxel.

Además de las métricas numéricas, este preprocesamiento generó varios productos intermedios. Para cada captura se guardó un mapa de profundidad corregido con ambos pies, así como una versión recortada y centrada de cada pie por separado. Este último paso se implementó identificando el contorno de cada pie, recortando un cuadro ajustado a su *bounding box* y embebiendo dicho recorte en una imagen cuadrada de  $1080 \times 1080$  píxeles, rellena con ceros en las zonas vacías. Las intensidades de profundidad se normalizaron al rango de 0 a 255 y se almacenaron en formato PNG. Estas imágenes normalizadas sirvieron más adelante como entrada al extractor de vectores de características (conocido en inglés como *embeddings*) basado en redes convolucionales.

Finalmente, todos los parámetros derivados del preprocesamiento (identificador de imagen, índice AHI, profundidad en el punto de arco, longitud entre antepié y talón en milímetros, área proyectada y volumen estimado bajo el arco) se guardaron en un archivo CSV de manera incremental. Este archivo funcionó como base de datos estructurada para el análisis posterior, tanto en la parte descriptiva de las características morfológicas como en la construcción de los modelos de clasificación empleados en los capítulos siguientes.

## 4.6. Área bajo la curva

En esta etapa se buscó una medida que integrara de manera sencilla la relación entre el volumen plantar y la superficie de apoyo del pie. Para ello se definió un Factor Standard, calculado como la razón entre el volumen proyectado bajo la curva plantar (obtenido a partir de la suma de profundidades relativas) y el área total de apoyo en milímetros cuadrados. Este cociente permite normalizar el volumen con respecto al tamaño del pie, lo cual resulta útil para comparar individuos de diferentes longitudes y anchos sin que la magnitud del pie influya directamente en el análisis.

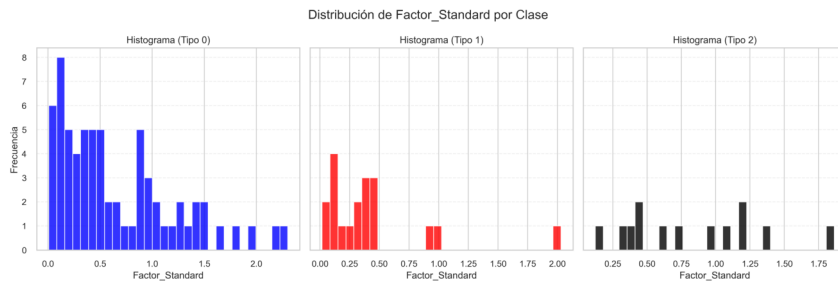
Tabla 4.2: Pruebas de normalidad aplicadas al Factor Standard en cada grupo.

Tipo	n	Shapiro-Wilk	p-valor	Normal?	K-S	p-valor	Normal?	A-D	Crítico (0.05)
0	68	0.8948	3.11E-05	No	0.1480	0.0915	Sí	2.1656	0.747
1	19	0.7137	8.13E-05	No	0.3129	0.0376	No	1.8018	0.690
2	13	0.9534	0.651	Sí	0.1456	0.909	Sí	0.2845	0.679

Posteriormente, se evaluó la distribución del Factor Standard en cada grupo, se aplicaron diferentes pruebas de normalidad con el fin de corroborar los resultados de manera más sólida. En la Tabla 4.2 se presentan los valores obtenidos mediante Shapiro-Wilk, Kolmogorov-Smirnov y Anderson-Darling. Para el grupo Tipo 0 ( $n=68$ ), las pruebas de Shapiro-Wilk y Anderson-Darling rechazaron la hipótesis de normalidad ( $p \leq 0.05$ ), mientras que Kolmogorov-Smirnov sugirió lo contrario; sin embargo, debido al tamaño de la muestra y a la sensibilidad de Shapiro-Wilk, se consideró que la distribución de este grupo no sigue un comportamiento normal. En el caso de Tipo 1 ( $n=19$ ), las tres pruebas coincidieron en descartar la normalidad, por lo que el resultado es consistente. Finalmente, para Tipo 2 ( $n=13$ ), los valores de  $p$  superaron el umbral de significancia y los estadísticos se mantuvieron por debajo de los puntos críticos, lo que permitió aceptar la normalidad en este grupo.

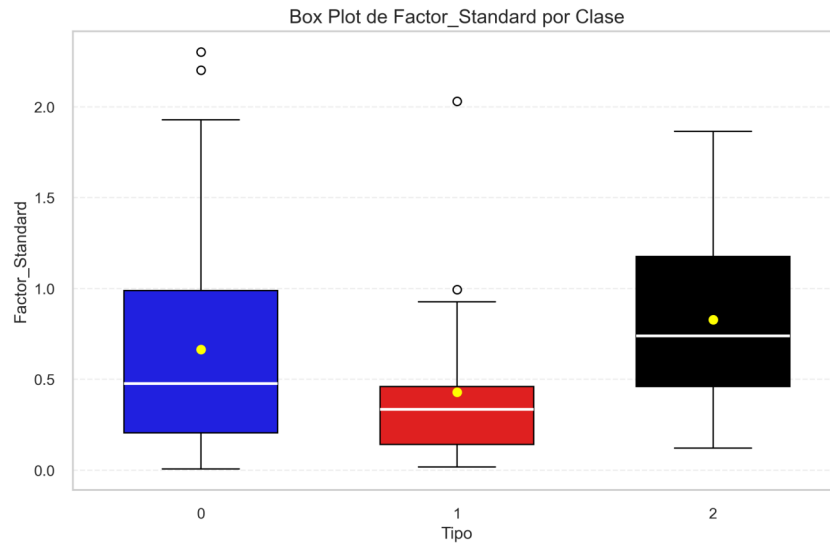
Con el propósito de complementar los resultados estadísticos, se representó gráficamente la distribución del Factor Standard mediante histogramas, diagramas de cajas y gráficos Q-Q. En los histogramas ver Figura 4.7, se observa que los datos de los grupos Tipo 0 y Tipo 1 presentan una asimetría evidente y una acumulación de

valores hacia los intervalos inferiores, lo que respalda la conclusión de que no siguen una distribución normal. El grupo Tipo 2, en cambio, muestra una distribución más equilibrada y sin concentraciones marcadas en los extremos.



**Figura 4.7:** *Histogramas por clase de pie.*

Los diagramas de caja permiten visualizar de forma comparativa la dispersión de los valores y la presencia de atípicos (ver Figura 4.8 ). El grupo Tipo 0 presenta mayor variabilidad, mientras que el Tipo 1 concentra la mayoría de sus valores en rangos reducidos y con menos amplitud intercuartílica. En el caso de Tipo 2, se aprecia una distribución más uniforme, en concordancia con lo indicado por las pruebas de normalidad.



**Figura 4.8:** *Boxplot por clase de pie.*

Finalmente, los gráficos Q-Q confirman estas observaciones: en los grupos Tipo 0 y Tipo 1 los puntos se desvían de la diagonal teórica de normalidad, especialmente en los extremos, mientras que en el Tipo 2 los valores se alinean de manera más cercana a la recta, sugiriendo un ajuste más adecuado a la distribución normal (ver Figura 4.9).



**Figura 4.9:** *QQ plot por clase de pie.*

En conjunto, estas representaciones gráficas corroboran lo obtenido en la tabla de pruebas estadísticas, mostrando que la mayoría de los grupos no cumplen con el supuesto de normalidad y sugiriendo la necesidad de emplear métodos estadísticos adecuados para este tipo de distribución.

Bajo este contexto, se evaluó el comportamiento del análisis de varianza (ANOVA) considerando sus supuestos básicos. La independencia de las observaciones se asumió debido a que cada medición corresponde a un individuo distinto, por lo que no existe relación entre las muestras. Asimismo, la homogeneidad de varianzas fue verificada mediante la prueba de Levene, obteniendo un valor de  $p = 0.2429$ , lo que indica que este supuesto se cumple.

Sin embargo, dado que la normalidad no se satisface en todos los grupos, el ANOVA no resulta completamente adecuado como prueba principal. Aun así, se aplicó como referencia comparativa, obteniendo un valor de  $p = 0.1044$ , lo que indica que bajo este enfoque no se detectan diferencias significativas entre los grupos.

En contraste, al emplear la prueba no paramétrica de Kruskal–Wallis, la cual no requiere el supuesto de normalidad, se obtuvo un estadístico de 6.37 con un valor de  $p = 0.041$  (véase Tabla 4.3). Este resultado es menor al umbral de significancia establecido ( $\alpha = 0.05$ ), lo que permite rechazar la hipótesis nula de igualdad de medianas. En consecuencia, puede afirmarse que existen diferencias estadísticamente significativas en la distribución de los valores del Factor Standard entre los tipos de pie, por lo que resulta pertinente realizar pruebas post hoc que permitan identificar entre qué grupos específicos se presentan dichas diferencias.

Tabla 4.3: Resultado de la prueba de Kruskal-Wallis aplicada al *Factor Standard* en los tres grupos de estudio.

Test	Statistic	p-valor	Significativo ( $\alpha=0.05$ )
Kruskal-Wallis	6.3795	0.0412	Sí

La Tabla 4.4 presenta los resultados de la prueba post hoc de Dunn con corrección de Bonferroni, aplicada tras detectar diferencias globales significativas mediante

Kruskal-Wallis. Se observa que la comparación entre los grupos Tipo 1 y Tipo 2 resultó significativa ( $p$  ajustado = 0.039 < 0.05), lo que indica que el comportamiento del Factor Standard difiere entre estos dos tipos de pie. En contraste, las comparaciones Tipo 0 vs. Tipo 1 y Tipo 0 vs. Tipo 2 no alcanzaron significancia estadística después de la corrección, aún cuando los valores de  $Z$  sugieren cierta tendencia a la diferenciación. Este resultado confirma que las diferencias detectadas en la prueba global se deben principalmente a la separación entre los grupos 1 y 2, lo que aporta evidencia de que el Factor Standard discrimina mejor entre estos casos que entre los grupos que involucran al Tipo 0.

Tabla 4.4: Resultados de la prueba post hoc de Dunn con corrección de Bonferroni para el *Factor Standard*.

Grupo A	Grupo B	n_A	n_B	MeanRank_A	MeanRank_B	Z	p sin ajuste	p Bonferroni	Signif. (0.05)
1	2	19	13	38.000	63.923	-2.4825	0.0130	0.0391	Sí
0	1	68	19	51.426	38.000	1.7835	0.0745	0.2235	No
0	2	68	13	51.426	63.923	-1.4230	0.1547	0.4642	No

## 4.7. Experimento 1: Clasificador Bayesiano con KDE sobre el Factor Standard

### 4.7.1. Desarrollo del método

#### 4.7.1.1. Planteamiento general

El propósito de este experimento fue evaluar la capacidad discriminativa del *Factor Standard* como única característica para la clasificación de huellas plantares en tres categorías: **Tipo 0 (normal)**, **Tipo 1 (plano)** y **Tipo 2 (cavo)**. Para ello se implementó un clasificador bayesiano donde la verosimilitud  $p(x | y = k)$  se modeló mediante un estimador no paramétrico KDE con núcleo gaussiano.

#### 4.7.1.2. Datos y particionado

A partir del archivo CSV general se seleccionaron únicamente las columnas `Factor_Standard` y `Tipo`. La base presenta un desbalance inherente (Tipo 0: 68; Tipo 1: 19; Tipo 2: 13), el cual se mantuvo mediante un particionado estratificado en tres subconjuntos: entrenamiento, validación y prueba. Esta estrategia asegura consistencia en la proporción de clases, aunque no modifica el desbalance estructural.

#### 4.7.1.3. Modelado KDE por clase

Para cada clase  $k \in \{0, 1, 2\}$ , la verosimilitud se estimó mediante:

$$\hat{p}(x | y = k) = \frac{1}{n_k h} \sum_{i=1}^{n_k} \mathcal{K}\left(\frac{x - x_i^{(k)}}{h}\right), \quad (4.4)$$

donde  $\mathcal{K}$  es un núcleo gaussiano y  $h = 1.0$  es el parámetro de suavizado empleado de manera uniforme en todas las clases. Este enfoque permite aproximar la distribución empírica del *Factor\_Standard* sin asumir normalidad.

#### 4.7.1.4. Regla bayesiana de decisión

Los priors se definieron a partir de la frecuencia empírica de cada clase en el conjunto de entrenamiento:

$$P(y = k) = \frac{n_k}{N}. \quad (4.5)$$

La probabilidad posterior se calculó mediante:

$$P(y = k | x) = \frac{p(x | y = k) P(y = k)}{\sum_j p(x | y = j) P(y = j)}. \quad (4.6)$$

La predicción final corresponde a:

$$\hat{y}(x) = \arg \underset{k}{\text{máx}} P(y = k | x). \quad (4.7)$$

Este esquema permite evaluar directamente si el *Factor\_Standard* proporciona suficiente capacidad de separación para un modelo bayesiano simple bajo un escenario de desbalance.

## 4.8. Experimento 2: Truncado balanceado (13/13/13) y Bayes-KDE

### 4.8.1. Planteamiento y objetivo

Con el fin de eliminar el sesgo por desbalance de clases sin introducir muestras sintéticas, se implementó un **balanceo por truncado**: para cada clase  $y \in \{0, 1, 2\}$  se conservaron únicamente **13 observaciones** (la clase minoritaria marca el tope), quedando un conjunto equilibrado de **39 muestras**.

El objetivo de este experimento es **aislar el efecto del desbalance** y evaluar si, con priors equilibrados y densidades estimadas con el mismo tamaño muestral por clase, el *Factor\_Standard* permite separar los grupos mediante un clasificador Bayes-KDE.

### 4.8.2. Datos y particionado

Antes del truncado, la distribución de clases en el conjunto original era: **Tipo 0 = 68, Tipo 1 = 19, Tipo 2 = 13**. Tras el truncado 13/13/13 se obtuvo: **Tipo 0 = 13, Tipo 1 = 13, Tipo 2 = 13**.

Sobre este conjunto balanceado se aplicó particionado estratificado con `test_size = 0.30` y `random_state = 42`, dando lugar a: **Entrenamiento**: 6 muestras por clase, **Validación**: 3 muestras por clase, **Prueba**: 4 muestras por clase.

Tabla 4.5: Distribución por clase antes y después del truncado, y por partición.

Conjunto	Clase 0	Clase 1	Clase 2	Total
Original	68	19	13	100
Truncado (13/13/13)	13	13	13	39
Train	6	6	6	18
Validación	3	3	3	9
Test	4	4	4	12

### 4.8.3. Modelado KDE por clase

La verosimilitud  $\hat{p}(x | y = k)$  se estimó con *Kernel Density Estimation* (KDE) gaussiano y ancho de banda fijo  $h = 1.0$ :

$$\hat{p}(x | y = k) = \frac{1}{n_k h} \sum_{i=1}^{n_k} \mathcal{K}\left(\frac{x - x_i^{(k)}}{h}\right), \quad \mathcal{K}(u) = \frac{1}{\sqrt{2\pi}} e^{-u^2/2}, \quad (4.8)$$

donde  $n_k$  es el número de muestras de la clase  $k$  en el conjunto de entrenamiento y  $\mathcal{K}$  es un núcleo gaussiano estándar. Este esquema permite aproximar la distribución empírica de `Factor_Standard` por clase sin asumir un modelo paramétrico específico. El análisis de las curvas KDE resultantes se presenta en el capítulo de Resultados.

#### 4.8.4. Regla bayesiana

La decisión se implementó mediante un clasificador Bayesiano sobre las densidades KDE estimadas. La regla de decisión se define como:

$$\hat{y}(x) = \arg \max_{k \in \{0,1,2\}} \pi_k \hat{p}(x | y = k), \quad (4.9)$$

donde  $\pi_k$  son los priors de clase calculados empíricamente en el conjunto de entrenamiento. Debido al truncado balanceado, estos priors resultan equiparables entre clases, lo que permite evaluar el comportamiento del modelo en un escenario con priors aproximadamente uniformes. El desempeño de este clasificador se describe en el capítulo de Resultados.

### 4.9. Experimento 3: KDE + Bayes con clases 0 y 1

#### 4.9.1. Planteamiento y objetivo

En este experimento se restringió el análisis únicamente a las clases **0** y **1**, con el fin de evaluar el desempeño del clasificador Bayesiano con estimación no paramétrica mediante KDE en un escenario binario.

La motivación principal fue analizar si, al eliminar la clase 2 del problema original, el modelo mejora su capacidad para detectar la clase minoritaria (Tipo 1) cuando sólo se dispone del `Factor_Standard` como característica.

### 4.9.2. Distribución de datos

Se utilizó el mismo conjunto de datos de huellas plantares que en los experimentos previos, filtrando únicamente los registros con `Tipo = 0` y `Tipo = 1`. La distribución original quedó de la siguiente manera:

- Clase 0: 68 muestras
- Clase 1: 19 muestras

Posteriormente se aplicó un particionado estratificado (70 % entrenamiento, 30 % prueba) incluyendo un subconjunto de validación, obteniéndose:

- **Entrenamiento:** 33 muestras de la clase 0 y 9 de la clase 1
- **Validación:** 14 muestras de la clase 0 y 4 de la clase 1
- **Prueba:** 21 muestras de la clase 0 y 6 de la clase 1

Este esquema mantiene la proporción entre clases en cada partición, preservando el desbalance original entre `Tipo 0` y `Tipo 1`.

### 4.9.3. Estimación KDE por clase

Para cada una de las clases consideradas ( $y \in \{0, 1\}$ ) se ajustó un modelo de *Kernel Density Estimation* (KDE) con núcleo gaussiano sobre el `Factor_Standard`. De forma análoga a los experimentos anteriores, la verosimilitud condicional se modeló como la ecuación (4.8).

El análisis visual de las curvas KDE y su solapamiento se presenta en el capítulo de Resultados.

#### 4.9.4. Regla bayesiana de decisión

Sobre las densidades estimadas se implementó un clasificador Bayesiano binario. La regla de decisión se definió como la ecuación (4.9) donde  $\pi_k$  son los priors empíricos calculados a partir de la frecuencia de cada clase en el conjunto de entrenamiento.

Este esquema permite evaluar el impacto combinado del desbalance entre clases y del solapamiento entre las densidades KDE en un escenario de dos clases basado únicamente en el `Factor_Standard`.

### 4.10. Experimento 4: MLP con doble particionado y particionado simple

#### 4.10.1. Planteamiento y objetivo

El objetivo de este experimento fue evaluar el desempeño de un perceptrón multicapa (MLP) aplicado a las matrices de profundidad aplanadas, considerando dos esquemas de particionado.

- **Parte 1:** doble particionado estratificado (70 % train+val / 30 % test, y dentro del 70 %, una división 70/30 para train y validación).
- **Parte 2:** particionado simple 70/30 (train/test).

La comparación entre ambos busca analizar el efecto de disponer o no de un conjunto de validación explícito sobre el aprendizaje y la estabilidad del modelo.

### 4.10.2. Datos y particionado

Se emplearon 100 muestras con una distribución desbalanceada entre clases: 68 % clase 0, 19 % clase 1 y 13 % clase 2. Los particionados fueron:

- **Parte 1 (doble split):** 49 muestras para entrenamiento, 21 para validación y 30 para prueba.
- **Parte 2 (split simple):** 70 muestras para entrenamiento y 30 para prueba.

En ambos casos se mantuvo particionado estratificado para conservar las proporciones por clase.

### 4.10.3. Preprocesamiento y aplanamiento de matrices

Cada archivo CSV de profundidad, con dimensiones originales  $1079 \times 1919$ , se aplanó en un vector unidimensional de más de dos millones de características.

El preprocesamiento fue idéntico en ambas partes:

- sustitución de valores `NaN` por ceros,
- normalización mediante `StandardScaler`,
- almacenamiento de los vectores en un arreglo final de entrenamiento.

Este procedimiento permitió garantizar comparabilidad entre ambos esquemas.

#### 4.10.4. Modelo MLP y parámetros de entrenamiento

Antes de definir la arquitectura final del perceptrón multicapa, se realizaron algunas pruebas exploratorias modificando distintos hiperparámetros del modelo, entre ellos el número de capas ocultas, la cantidad de neuronas por capa, las funciones de activación y el optimizador. Estas pruebas permitieron observar el comportamiento del modelo durante el entrenamiento y verificar qué configuraciones mostraban una convergencia más estable.

Con base en estas pruebas preliminares se eligió la configuración que presentó un desempeño más consistente durante el entrenamiento. La arquitectura final utilizada en este experimento fue la siguiente:

- tres capas ocultas de 128 neuronas cada una,
- función de activación ReLU,
- optimizador Adam,
- máximo de 100 épocas.

La diferencia metodológica entre ambas partes fue:

- **Parte 1:** entrenamiento sobre train y evaluación preliminar en validación antes de probar en test.
- **Parte 2:** entrenamiento directo sobre train y evaluación únicamente en test. Se registró adicionalmente la curva de pérdida durante el entrenamiento.

Los resultados obtenidos mediante ambas configuraciones se presentan en el capítulo de Resultados.

## 4.11. Experimento 5: MLP sobre la matriz de profundidad de todo el pie

### 4.11.1. Planteamiento y objetivo

El propósito de este experimento fue evaluar el desempeño de un perceptrón multicapa (MLP) aplicado sobre las matrices de profundidad **completas** del pie (incluyendo huella y arco). A diferencia de los Experimentos 1–4, donde sólo se utilizaban valores asociados al *volumen bajo el arco*, aquí se consideró la información de todo el pie.

Se diseñaron cuatro configuraciones:

- **Parte 1 (tres clases):** Pie derecho, clasificación en  $\{0, 1, 2\}$ .
- **Parte 2 (dos clases):** Pie derecho, filtrando únicamente  $\{0, 1\}$ .
- **Parte 3 (tres clases, unificación de lado):** Conjunto aumentado a 200 muestras combinando pie derecho real + pie izquierdo espejado, con clases  $\{0, 1, 2\}$ .
- **Parte 4 (dos clases, unificación de lado):** Igual que la Parte 3, pero filtrando sólo clases  $\{0, 1\}$ , quedando 174 muestras.

El objetivo general es analizar el impacto de: (i) trabajar con tres vs. dos clases y (ii) unificar el lado del pie mediante espejeo, sobre el desempeño del MLP cuando se usan matrices de profundidad completas.

### 4.11.2. Datos y particionado

En todas las partes, cada archivo CSV de dimensiones  $1080 \times 500$  se aplanó a un vector de 540 000 atributos. Se reemplazaron valores NaN por 0 y se aplicó normalización mediante `StandardScaler`.

Los tamaños de muestra y particionados fueron:

- **Parte 1:** 100 muestras (clases 68/19/13). Split estratificado 70/30  $\Rightarrow$  70 entrenamiento, 30 prueba.
- **Parte 2:** 87 muestras (68/19; sólo clases 0 y 1). Split estratificado 70/30  $\Rightarrow$  60 entrenamiento, 27 prueba.
- **Parte 3:** 200 muestras (136/38/26; pie derecho + pie izquierdo espejado). Split estratificado 70/30  $\Rightarrow$  140 entrenamiento, 60 prueba.
- **Parte 4:** 174 muestras (136/38; sólo clases 0 y 1). Split estratificado 70/30  $\Rightarrow$  121 entrenamiento, 53 prueba.

### 4.11.3. Modelo MLP y parámetros de entrenamiento

Al igual que en el Experimento 4, se realizaron pruebas exploratorias variando distintos hiperparámetros del perceptrón multicapa, como el número de capas ocultas, la función de activación, el optimizador y el número de iteraciones. A partir de estas configuraciones se seleccionó la que mostró el mejor desempeño durante el entrenamiento. Esta misma configuración se mantuvo entre los Experimentos 4 y 5, lo cual facilita la comparación entre ambos y permite evaluar con mayor claridad el impacto del tipo de entrada (volumen bajo el arco frente a matriz completa de profundidad).

En las cuatro partes se utilizó un *pipeline* de entrenamiento implementado en `scikit-learn`, compuesto por una etapa de normalización mediante `StandardScaler` seguida del clasificador `MLPClassifier`.

- tres capas ocultas de 128 neuronas,
- función de activación ReLU,
- optimizador Adam,
- máximo de 100 épocas.

En todos los casos se registró la curva de pérdida durante el entrenamiento, con el fin de analizar el comportamiento de convergencia del modelo. Las curvas correspondientes se presentan y analizan en el capítulo de Resultados.

## 4.12. Experimento 6: Extracción de vectores de características

### 4.12.1. Planteamiento y objetivo

El objetivo de este experimento fue: (i) construir un extractor de vectores de características para huellas plantares a partir de una CNN preentrenada (VGG-16) y (ii) evaluar, sobre dichos vectores de características, varios clasificadores clásicos para la detección binaria **Plano** vs. **No plano**.

La hipótesis de trabajo es que la representación latente de una CNN (penúltima capa) captura rasgos morfológicos discriminativos que facilitan la separación de clases utilizando modelos de baja varianza (LogReg, SVM lineal, kNN, ANN/MLP y Random Forest).

### 4.12.2. Datos y particionado

Se trabajó con **200 pies** (100 izquierdos y 100 derechos). Cada pie proviene de una **matriz de profundidad** (medición 3D); a cada uno se le asignó un **ID** único. Para poder aprovechar arquitecturas 2D, cada volumen 3D se proyectó a una **imagen 2D** en la que la intensidad codifica la profundidad.

Las etiquetas originales en el CSV (*Tipo/Clase*) eran:

$$\{0 : \text{No plano}, 1 : \text{Plano}, 2 : \text{Cavo}\}.$$

Para este experimento se redujeron a dos clases mediante el mapeo:

$$2 (\text{Cavo}) \rightarrow 0 (\text{No plano}),$$

quedando la tarea binaria:

$$\{0 : \text{No plano}, 1 : \text{Plano}\}.$$

El particionado se realizó de forma **estratificada 80/20** a nivel de imagen:

- **Train:**  $n = 160$  imágenes (129 No plano, 31 Plano).
- **Validación:**  $n = 40$  imágenes (32 No plano, 8 Plano).

Globalmente, el conjunto presenta 161 pies No plano (80.5%) y 39 pies Plano (19.5%), lo que implica un desbalance aproximado 80/20.

### 4.12.3. Modelos y parámetros de entrenamiento

**Extractor de vectores de características (Código A).** Se utilizó **VGG-16** preentrenada en ImageNet con `include_top=false` y **pesos congelados**. Sobre la salida convolucional se añadió la siguiente “cabeza” densa:

Flatten  $\rightarrow$  Dense(256, ReLU) [embed\_256]  $\rightarrow$  Dropout(0.5)  $\rightarrow$  Dense(1, Sigmoid).

Los principales detalles de entrenamiento fueron:

- Optimizador: **Adam** con tasa de aprendizaje  $lr = 10^{-4}$ .
- Tamaño de lote: **batch** = 16.
- Número de épocas: 10.
- Tamaño de imagen:  $224 \times 224$ .
- Reescalado de intensidades: factor  $1/255$ .
- Callbacks: **EarlyStopping** y **ModelCheckpoint** monitorizando `val_loss`.

Se extrajeron dos tipos de *vectores de características*:

- **Dense-256**: salida de la capa `embed_256` (vector 256D, especializado en la tarea).
- **GAP-512**: salida de una capa `GlobalAveragePooling2D` aplicada sobre la salida convolucional de VGG-16 (vector 512D, más genérico).

Para este experimento se consideró principalmente **Dense-256** como representación de entrada para los clasificadores clásicos.

**Clasificadores clásicos (Código B).** Sobre los **vectores de características Dense-256** se entrenaron los siguientes modelos:

- **Logistic Regression**: `class_weight='balanced'`, `max_iter=2000`, en un Pipeline con `StandardScaler`.

- **Linear SVM** (LinearSVC): `class_weight='balanced'` + StandardScaler.
- **kNN**:  $k = 9$ , `weights='distance'` + StandardScaler.
- **ANN** (MLPClassifier): arquitectura (128, 64), activación ReLU, `early_stopping=true`, `max_iter=400` + StandardScaler.
- **Random Forest**: `n_estimators = 300`, `class_weight='balanced'`.

El desempeño de estos modelos sobre el conjunto de validación se describe en el capítulo de Resultados.

## 4.13. Experimento 7: Matriz de profundidad + vectores de características (pie derecho real)

### 4.13.1. Planteamiento y objetivo

En este experimento se buscó integrar dos tipos de información para la clasificación binaria **Plano** vs. **No plano**: (i) la **matriz de profundidad completa** del pie derecho real y (ii) los **vectores de características Dense-256** obtenidos previamente con la CNN del Experimento 6.

La hipótesis es que concatenar la geometría completa del mapa de profundidad con una representación latente compacta mejora la separabilidad entre clases, especialmente al optimizar la métrica **F1\_weighted** para mitigar el efecto del desbalance entre pies planos y no planos.

### 4.13.2. Datos y particionado

Se utilizaron **100 pies derechos reales**, cada uno representado por una matriz de profundidad de tamaño  $1080 \times 500$ . A cada ID de pie derecho real (`ID.#_pie_derecho_real`) se le asoció además un vector de **vectores de características Dense-256** ( $e_0, \dots, e_{255}$ ) extraído del modelo VGG-16 del Experimento 6.

Para cada muestra se construyó un vector de características de dimensión:

$$540000 + 256 = 540256,$$

correspondiente a la matriz de profundidad aplanada concatenada con los 256 *vectores de características*.

Las etiquetas originales (Tipo) seguían la codificación:

$$\{0 : \text{No plano}, 1 : \text{Plano}, 2 : \text{Cavo}\}.$$

Como en el Experimento 6, se redefinió un problema binario mediante el mapeo:

$$2 (\text{Cavo}) \rightarrow 0 (\text{No plano}),$$

quedando dos clases:  $\{0 : \text{No plano}, 1 : \text{Plano}\}$ . La distribución global fue:

$$n = 100, \quad 81 \text{ No plano (81 \%)} \text{ y } 19 \text{ Plano (19 \%)},$$

es decir, un desbalance aproximado 80/20.

El particionado se realizó con un **split estratificado 70/30** a nivel de imagen:

- **Train** ( $n = 70$ ): 57 No plano y 13 Plano (81.43 % vs. 18.57 %).
- **Test** ( $n = 30$ ): 24 No plano y 6 Plano (80 % vs. 20 %).

Este particionado conserva el desbalance original, lo que hace especialmente relevante el uso de métricas ponderadas como **F1\_weighted** y la optimización explícita del desempeño en la clase minoritaria.

### 4.13.3. Modelos y parámetros de entrenamiento

Sobre el vector combinado [**matriz aplanada || vectores de características Dense-256**] se evaluaron cuatro clasificadores clásicos: **Logistic Regression**, **SVM lineal**, **kNN** y **Random Forest**. En todos los casos se empleó un Pipeline con **estandarización** previa (**StandardScaler**) para estabilizar el entrenamiento en alta dimensión.

Para cada modelo se realizó una búsqueda en rejilla (**GridSearchCV**) con **validación cruzada estratificada 5-fold** y métrica objetivo **F1\_weighted**, con el fin de encontrar hiperparámetros que penalicen lo menos posible la pérdida de la clase minoritaria (1: Plano).

#### Esquema de validación.

- **CV:** `StratifiedKFold(n_splits=5, shuffle=true, random_state=42)`.
- **Métrica objetivo:** `scoring='f1_weighted'`.
- **Búsqueda:** `GridSearchCV` con `return_train_score=true` para registrar la historia de `F1_weighted` en la rejilla.

#### Rejillas de hiperparámetros por modelo.

- **Logistic Regression (LogReg)**
  - Pipeline: `StandardScaler → LogisticRegression(max_iter=1000)`.
  - Rejilla:

$$C \in \{0.01, 0.1, 1, 10\}, \quad \text{class\_weight} \in \{\text{None}, \text{'balanced'}\}.$$

- **SVM lineal (LINEAR\_SVM)**

- Pipeline: `StandardScaler` → `SVC(kernel='linear')`.
- Rejilla:

$$C \in \{0.01, 0.1, 1, 10\}, \quad \text{class\_weight} \in \{\text{None}, \text{'balanced'}\}.$$

- **kNN**

- Pipeline: `StandardScaler` → `KNeighborsClassifier()`.
- Rejilla:

$$k \in \{3, 5, 7, 9\}, \quad \text{weights} \in \{\text{'uniform'}, \text{'distance'}\}, \quad p \in \{1, 2\}.$$

- **Random Forest (RF)**

- Pipeline: `StandardScaler` → `RandomForestClassifier(random_state=42)`.
- Rejilla:

$$\begin{aligned} n\_estimators &\in \{100, 200, 300\}, \quad \text{max\_depth} \in \{\text{None}, 10, 20\}, \\ \text{min\_samples\_split} &\in \{2, 5\}, \quad \text{class\_weight} \in \{\text{None}, \text{'balanced'}\}. \end{aligned}$$

Las trayectorias de `F1_weighted` a lo largo de las configuraciones de la rejilla para cada modelo se presentan y analizan en el capítulo de Resultados.

## 4.14. Experimento 8: vectores de características con balanceo mediante SMOTE

### 4.14.1. Planteamiento y objetivo

El objetivo de este experimento fue evaluar el desempeño de varios clasificadores clásicos entrenados *exclusivamente* sobre *vectores de características* de huellas plantares, incorporando un esquema explícito de **rebalanceo de clases** mediante **SMOTE** (*Synthetic Minority Over-sampling Technique*).

A diferencia de experimentos previos, donde se combinaba la matriz de profundidad aplanada con *vectores de características* o se trabajaba directamente sobre matrices de profundidad de alta dimensión, aquí se busca explotar únicamente la representación latente generada por la CNN (Dense-256 del Experimento 7) y analizar si el balanceo sintético mejora tanto la sensibilidad a **Plano** como las métricas globales en una tarea binaria **No plano** vs. **Plano**. La hipótesis es que, sobre un espacio de características compacto (256D) y con oversampling controlado en la clase minoritaria, los modelos de baja varianza (LogReg, SVM lineal, kNN y Random Forest) pueden alcanzar un buen compromiso entre *accuracy*, F1 y *recall* de la clase positiva.

#### 4.14.2. Datos y particionado

Se trabajó con **100 huellas de pie derecho real**, cada una representada por un vector de características de dimensión 256 (`e0--e255`), previamente calculado a partir de la arquitectura VGG-16 usada en el Experimento 6. Cada huella está asociada a un identificador único `ID_x_pie_derecho_real.png` y a una etiqueta *Tipo* en el CSV de parámetros.

Las clases originales eran:

0 : No plano,    1 : Plano,    2 : Cavo.

Como en el Experimento 6, se realizó el mapeo:

2 (Cavo)  $\rightarrow$  0 (No plano),

obteniendo así una tarea binaria:

{0 : No plano (incluye Cavo),    1 : Plano}.

La distribución global después del mapeo fue de 81 muestras No plano (81 %) y 19 muestras Plano (19 %), lo que mantiene un **desbalance 80/20** similar al observado en experimentos anteriores.

El particionado fue **estratificado 70/30**:

- **Train:**  $n = 70$  (57 No plano, 13 Plano).
- **Test:**  $n = 30$  (24 No plano, 6 Plano).

Sobre el conjunto de entrenamiento se aplicó **SMOTE** para generar muestras sintéticas de la clase minoritaria (Plano), aumentando su cardinalidad hasta aproximarla a la de la clase mayoritaria. La Tabla 4.6 resume el cambio en la distribución de clases en *train* antes y después de SMOTE.

Tabla 4.6: Distribución de clases en el conjunto de entrenamiento antes y después de SMOTE (Experimento 9).

Estado	Clase 0 (No plano)	Clase 1 (Plano)	Total
Antes de SMOTE	57	13	70
Después de SMOTE	57	57	114

#### 4.14.3. Modelos y parámetros de entrenamiento

En este experimento los atributos de entrada son **exclusivamente** los *vectores de características* Dense-256 (256 características por muestra). Sobre este espacio se entrenaron cuatro clasificadores clásicos, todos organizados en una **Pipeline** común:

SMOTE → StandardScaler → Clasificador.

El entrenamiento se realizó mediante **GridSearchCV** con validación cruzada estratificada de 5 folds, usando como métrica de optimización  $F1_{\text{weighted}}$ .

**Clasificadores clásicos y rejillas de hiperparámetros.** Se consideraron los siguientes modelos y rejillas:

- **Logistic Regression (LOGREG)**

- Base: `LogisticRegression(max_iter=1000, n_jobs=-1)`.
- Rejilla:

$$C \in \{0.01, 0.1, 1, 10\},$$
$$\text{class\_weight} \in \{\text{None}, \text{'balanced'}\}.$$

- **SVM lineal (LINEAR\_SVM)**

- Base: `SVC(kernel='linear', probability=false)`.
- Rejilla:

$$C \in \{0.01, 0.1, 1, 10\},$$
$$\text{class\_weight} \in \{\text{None}, \text{'balanced'}\}.$$

- **k-Nearest Neighbors (KNN)**

- Base: `KNeighborsClassifier()`.
- Rejilla:

$$k \in \{3, 5, 7, 9\},$$
$$\text{weights} \in \{\text{'uniform'}, \text{'distance'}\}.$$

- **Random Forest (RF)**

- Base: `RandomForestClassifier(random_state=42, n_jobs=-1)`.
- Rejilla:

$$n_{\text{estimators}} \in \{100, 200, 500\},$$
$$\text{max\_depth} \in \{\text{None}, 10, 20\},$$
$$\text{min\_samples\_split} \in \{2, 5\}.$$

En todos los casos, el **rebalanceo mediante SMOTE** se aplica exclusivamente al conjunto de entrenamiento dentro de cada fold de la validación cruzada. El conjunto de prueba (30 muestras) se mantiene intacto para obtener una evaluación honesta del desempeño final, reportada en el capítulo de Resultados.

---

## CAPÍTULO 5

---

# Resultados

### Contenido

---

<b>5.1. Resultados del Experimento 1 . . . . .</b>	<b>99</b>
5.1.1. Densidades KDE por clase . . . . .	99
5.1.2. Desempeño en validación y prueba . . . . .	100
5.1.3. Métricas de desempeño . . . . .	101
<b>5.2. Resultados del Experimento 2: Truncado 13/13/13 +     Bayes–KDE . . . . .</b>	<b>101</b>
5.2.1. Densidades KDE tras truncado . . . . .	101
5.2.2. Matrices de confusión en validación y prueba . . . . .	102
5.2.3. Métricas de desempeño . . . . .	104
<b>5.3. Resultados del Experimento 3: KDE + Bayes con clases     0 y 1 . . . . .</b>	<b>104</b>
5.3.1. Densidades KDE en entrenamiento . . . . .	104
5.3.2. Matrices de confusión en validación y prueba . . . . .	105
5.3.2.1. Resultados en validación . . . . .	106
5.3.2.2. Resultados en prueba . . . . .	106
<b>5.4. Resultados del Experimento 4: MLP con doble particio-     nado y particionado simple . . . . .</b>	<b>107</b>
5.4.1. Comparación de métricas entre Partes 1 y 2 . . . . .	107
5.4.2. Desempeño detallado por parte . . . . .	107
5.4.2.1. Parte 1: doble particionado (train/val/test) . . . . .	107
5.4.2.2. Parte 2: particionado simple (train/test) . . . . .	108
<b>5.5. Resultados del Experimento 5: MLP sobre la matriz de     profundidad de todo el pie . . . . .</b>	<b>109</b>

---

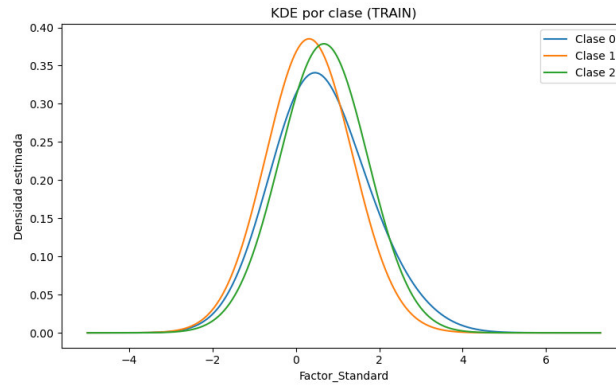
5.5.1. Curvas de pérdida y comportamiento del entrenamiento . . .	109
5.5.2. Métricas y comportamiento por parte . . . . .	112
<b>5.6. Resultados del Experimento 6: Embeddings . . . . .</b>	<b>113</b>
5.6.1. Métricas globales en validación . . . . .	113
5.6.2. Patrones en las matrices de confusión . . . . .	114
<b>5.7. Resultados del Experimento 7: Matriz de profundidad</b>	
<b>+ embeddings (pie derecho real) . . . . .</b>	<b>115</b>
5.7.1. Historia de F1_weighted en la búsqueda de hiperparámetros	115
5.7.2. Métricas y matrices de confusión en test . . . . .	117
<b>5.8. Resultados del Experimento 8: Embeddings con balan-</b>	
<b>ceo mediante SMOTE . . . . .</b>	<b>119</b>
5.8.1. Desempeño en validación (GridSearchCV) . . . . .	119
5.8.2. Métricas y matrices de confusión en prueba . . . . .	120

---

## 5.1. Resultados del Experimento 1

### 5.1.1. Densidades KDE por clase

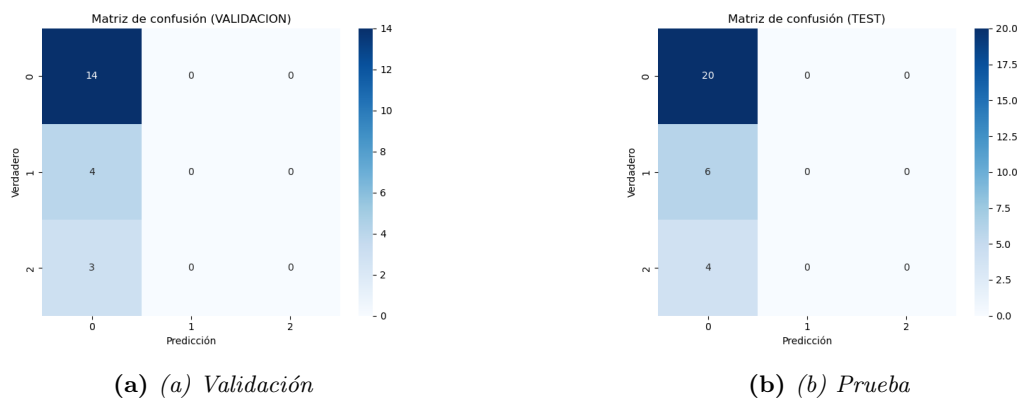
La Figura 5.1 muestra las densidades estimadas por clase en el conjunto de entrenamiento. Se observa un solapamiento considerable entre las distribuciones de los tres tipos, lo cual anticipa dificultades para lograr fronteras de decisión efectivas utilizando únicamente el *Factor\_Standard*.



**Figura 5.1:** Densidades KDE por clase con división estratificada.

### 5.1.2. Desempeño en validación y prueba

La Figura 5.2 presenta las matrices de confusión para los conjuntos de validación y prueba. En ambas particiones, el clasificador asigna prácticamente todos los ejemplos a la clase mayoritaria (Tipo 0), reflejando un colapso del modelo hacia el prior empírico.



**Figura 5.2:** Matrices de confusión en validación y prueba para el clasificador bayesiano con KDE.

### 5.1.3. Métricas de desempeño

La Tabla 5.1 resume las métricas de precisión, recall y F1-score. Aunque la *accuracy* alcanza aproximadamente un 66.7%, esto se debe únicamente al desbalance del conjunto: el modelo no logra identificar correctamente las clases 1 y 2, produciendo valores nulos de recall y F1 para ambas.

Tabla 5.1: Métricas de desempeño para validación y prueba.

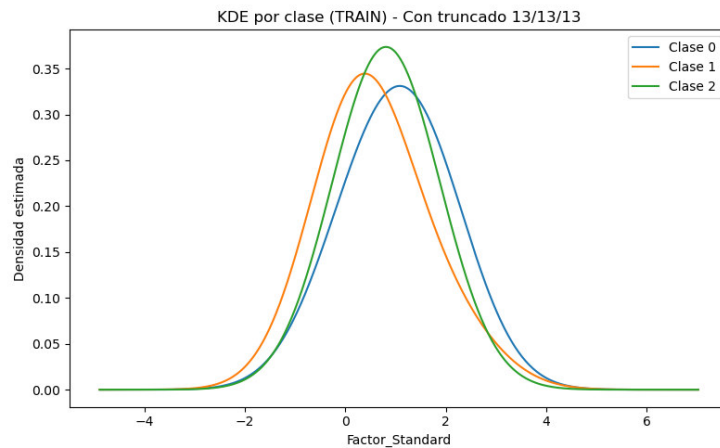
Conjunto	Clase	Precisión	Recall	F1-score	Soporte
Validación	0	0.6667	1.0000	0.8000	14
Validación	1	0.0000	0.0000	0.0000	4
Validación	2	0.0000	0.0000	0.0000	3
Validación (macro avg)	–	0.2222	0.3333	0.2667	21
Validación (weighted avg)	–	0.4444	0.6667	0.5333	21
Test	0	0.6667	1.0000	0.8000	20
Test	1	0.0000	0.0000	0.0000	6
Test	2	0.0000	0.0000	0.0000	4
Test (macro avg)	–	0.2222	0.3333	0.2667	30
Test (weighted avg)	–	0.4444	0.6667	0.5333	30

## 5.2. Resultados del Experimento 2: Truncado 13/13/13 + Bayes-KDE

### 5.2.1. Densidades KDE tras truncado

La Figura 5.3 muestra las densidades KDE estimadas por clase en el conjunto de entrenamiento, después de aplicar el truncado balanceado 13/13/13.

Se observa un **solapamiento importante** entre las distribuciones de las tres clases. La clase 1 tiende a desplazarse ligeramente hacia valores menores de `Factor_Standard`, mientras que las clases 0 y 2 se superponen en buena parte del soporte, lo que anticipa una separación limitada cuando se utiliza únicamente esta característica.

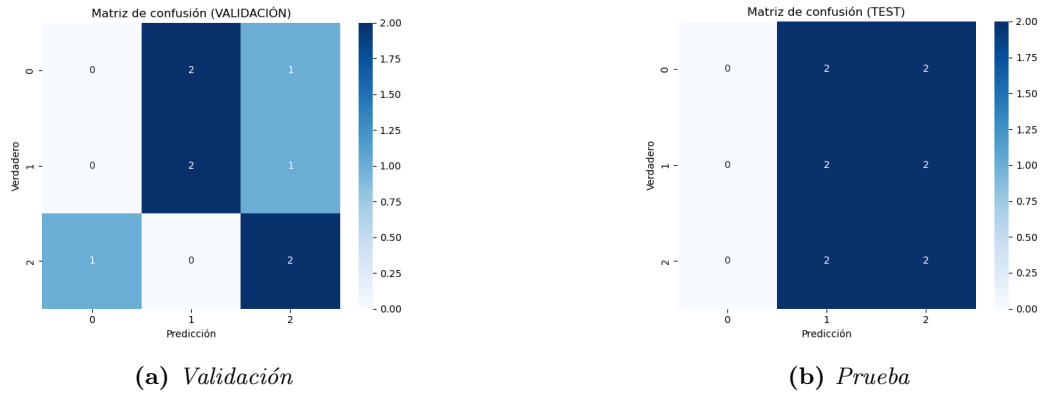


**Figura 5.3:** Densidades KDE por clase (TRAIN) tras rebalanceo por truncado 13/13/13.

### 5.2.2. Matrices de confusión en validación y prueba

La Figura 5.4 presenta las matrices de confusión obtenidas en los conjuntos de validación y prueba.

En validación, el modelo logra identificar de forma parcial las clases 1 y 2, mientras que la clase 0 no es recuperada. En prueba, el comportamiento se acentúa: la clase 0 no es predicha en ningún caso, y las decisiones se reparten entre las clases 1 y 2.



**Figura 5.4:** *Matrices de confusión en validación y prueba con rebalanceo por truncado 13/13/13.*

De manera explícita, las matrices de confusión para validación y prueba fueron:

**Validación** (9 observaciones; 3 por clase):

$$\begin{bmatrix} 0 & 2 & 1 \\ 0 & 2 & 1 \\ 1 & 0 & 2 \end{bmatrix}$$

La exactitud global fue de 0.444. Por clase:

- Clase 0: *recall* = 0.00.
- Clase 1: *precision* = 0.50, *recall* = 0.67.
- Clase 2: *precision* = 0.50, *recall* = 0.67.

**Prueba** (12 observaciones; 4 por clase):

$$\begin{bmatrix} 0 & 2 & 2 \\ 0 & 2 & 2 \\ 0 & 2 & 2 \end{bmatrix}$$

La exactitud fue de 0.333. En este caso, el modelo **no predice la clase 0** y reparte las decisiones entre las clases 1 y 2, ambas con  $recall = 0.50$ .

### 5.2.3. Métricas de desempeño

La Tabla 5.2 resume las métricas de desempeño en términos de *accuracy* y promedios macro de precisión, recall y F1-score para validación y prueba.

Tabla 5.2: Métricas de desempeño (macro-promedio) para el Experimento 2.

Conjunto	Accuracy	Macro-Precision	Macro-Recall	Macro-F1
Validación	0.4444	0.3333	0.4444	0.3810
Test	0.3333	0.2222	0.3333	0.2667

A pesar de contar con priors equilibrados, las métricas muestran un desempeño limitado: la *accuracy* cae por debajo del 50%, y las medidas macro de F1 reflejan la dificultad del modelo para separar de manera robusta las tres clases únicamente a partir de `Factor_Standard`.

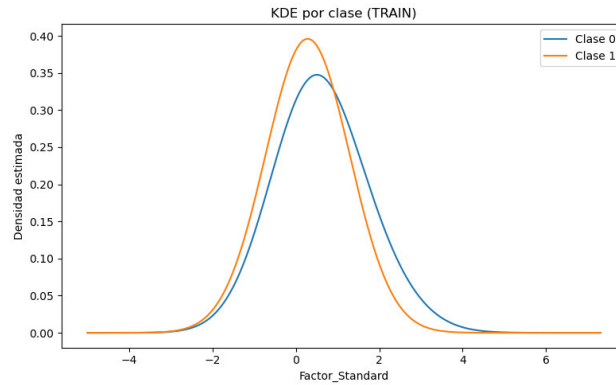
## 5.3. Resultados del Experimento 3: KDE + Bayes con clases 0 y 1

### 5.3.1. Densidades KDE en entrenamiento

La Figura 5.5 muestra las densidades KDE estimadas para las clases 0 y 1 en el conjunto de entrenamiento.

Las distribuciones presentan un solapamiento considerable, con ligeras diferencias de concentración alrededor del valor central de `Factor_Standard`. Sin embargo,

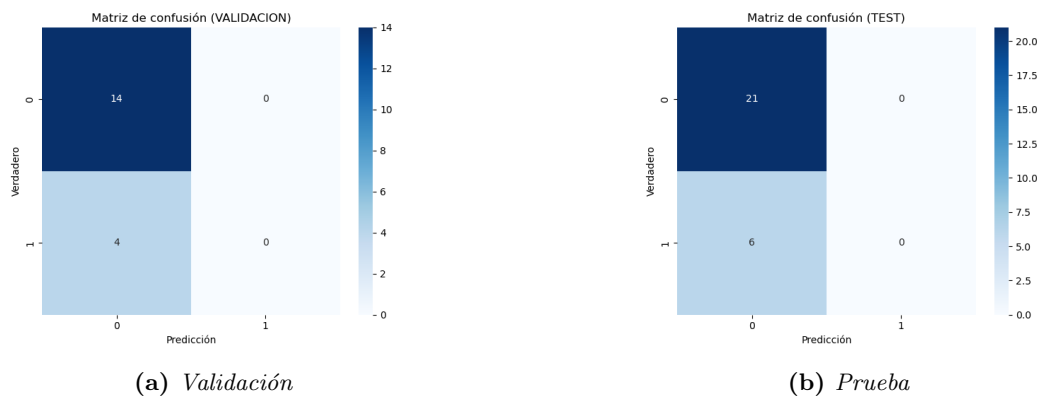
dicha separación resulta limitada para discriminar adecuadamente entre ambas clases únicamente con esta característica.



**Figura 5.5:** Densidades KDE por clase (TRAIN) para clases 0 y 1.

### 5.3.2. Matrices de confusión en validación y prueba

La Figura 5.6 presenta las matrices de confusión obtenidas para los conjuntos de validación y prueba.



**Figura 5.6:** Matrices de confusión en validación y prueba para el Experimento 3.

### 5.3.2.1. Resultados en validación

En el conjunto de validación, la matriz de confusión (Figura 5.6a) muestra que **todas** las muestras son clasificadas como clase 0, sin aciertos en la clase 1.

El reporte de métricas correspondiente es:

- **Exactitud:** 77.7%
- **Clase 0:** precisión 0.77, recall 1.00, F1-score 0.87
- **Clase 1:** precisión 0.00, recall 0.00, F1-score 0.00

### 5.3.2.2. Resultados en prueba

En el conjunto de prueba (Figura 5.6b) se observa el mismo patrón: todas las predicciones corresponden a la clase 0, sin detección de la clase 1. Las métricas obtenidas fueron:

- **Exactitud:** 77.7%
- **Clase 0:** precisión 0.77, recall 1.00, F1-score 0.87
- **Clase 1:** precisión 0.00, recall 0.00, F1-score 0.00

En ambos subconjuntos, la exactitud aparente es elevada debido al desbalance de clases; sin embargo, la clase minoritaria permanece completamente invisible para el modelo.

## 5.4. Resultados del Experimento 4: MLP con doble particionado y particionado simple

### 5.4.1. Comparación de métricas entre Partes 1 y 2

La Tabla 5.3 resume las métricas obtenidas en los dos esquemas de particionado. Los resultados muestran desempeños similares en términos de accuracy y F1 ponderado, indicando que el cambio en la estrategia de particionado no produce variaciones drásticas en el comportamiento del modelo.

Tabla 5.3: Comparación de métricas entre las Partes 1 (doble split 70/21/30) y 2 (split simple 70/30) del Experimento 4.

Parte	Conjunto	Accuracy	F1-macro	F1-weighted
Parte 1	Validación	0.6190	0.5159	0.6111
Parte 1	Test	0.5333	0.3519	0.5391
Parte 2	Test	0.5667	0.3165	0.5293

En la Parte 1, el modelo alcanzó en validación un F1-macro de 0.52, desempeño que no se mantuvo posteriormente en prueba. En la Parte 2, el modelo reportó accuracy y F1 ponderado ligeramente mayores en comparación con la Parte 1, aunque sin mejoras sustanciales en la detección de clases minoritarias.

### 5.4.2. Desempeño detallado por parte

#### 5.4.2.1. Parte 1: doble particionado (train/val/test)

En validación, el modelo alcanzó:

- **Accuracy:** 0.6190,
- **F1-macro:** 0.52,
- buen reconocimiento de la clase 0 y desempeño limitado en las clases 1 y 2.

En prueba, el desempeño disminuyó:

- **Accuracy:** 0.5333,
- **F1-macro:** 0.35,
- la clase 2 no fue reconocida.

#### 5.4.2.2. Parte 2: particionado simple (train/test)

La curva de pérdida mostró una rápida convergencia, con valores muy bajos desde la iteración 8, lo cual indica un ajuste fuerte al conjunto de entrenamiento sin mejoras claras en generalización.

En el conjunto de prueba:

- **Accuracy:** 0.5667,
- **F1-macro:** 0.31,
- la clase mayoritaria fue detectada correctamente, mientras que la clase 2 no registró aciertos.

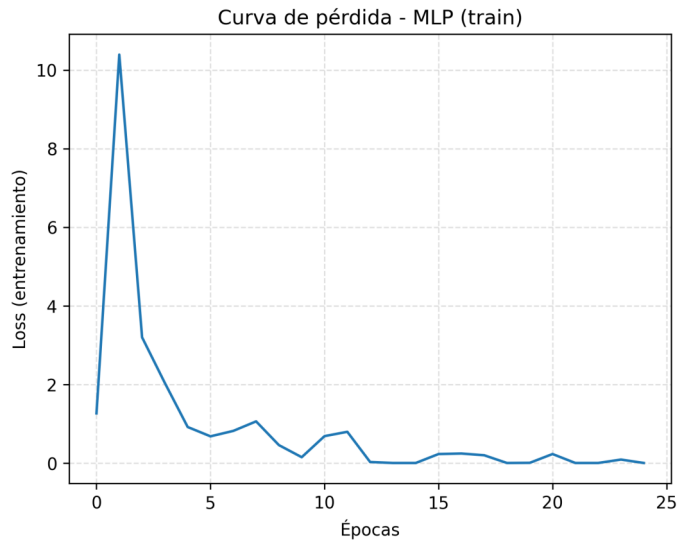
Estos resultados confirman que el desempeño entre ambas partes es similar y que el esquema de particionado no constituye el factor crítico del rendimiento.

## 5.5. Resultados del Experimento 5: MLP sobre la matriz de profundidad de todo el pie

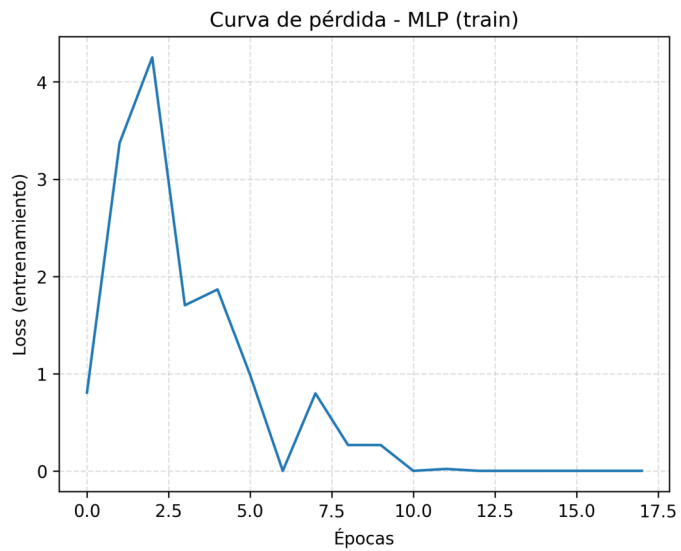
### 5.5.1. Curvas de pérdida y comportamiento del entrenamiento

En las cuatro partes del experimento se observó una rápida reducción de la función de pérdida durante las primeras épocas del entrenamiento. Las Figuras 5.7–5.10 muestran las curvas de pérdida para cada configuración:

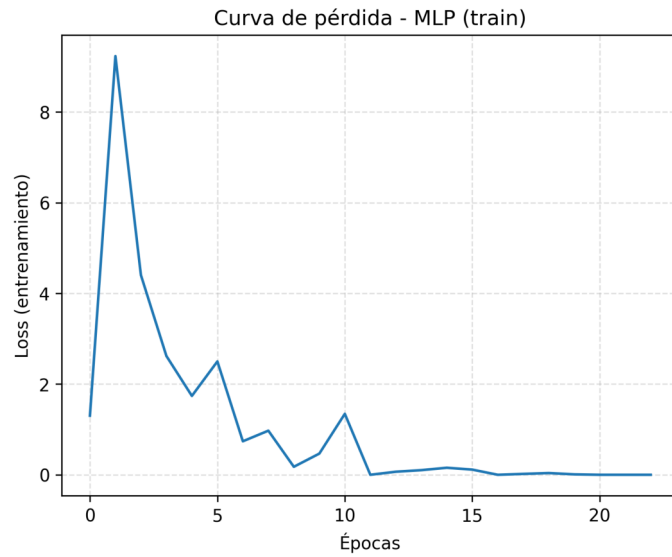
- **Parte 1:** convergencia alrededor de la época 25.
- **Parte 2:** convergencia alrededor de la época 18.
- **Parte 3:** convergencia alrededor de la época 23.
- **Parte 4:** convergencia alrededor de la época 25, con descenso particularmente rápido al inicio.



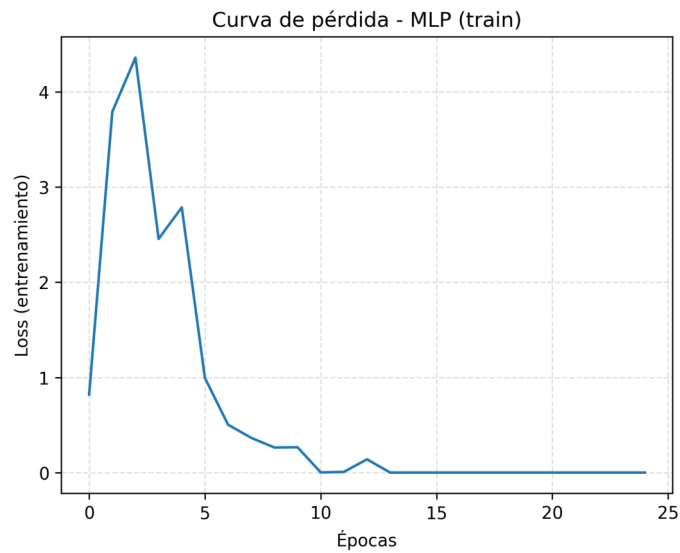
**Figura 5.7:** Curva de pérdida del MLP en el Experimento 5(1) (tres clases, pie derecho).



**Figura 5.8:** Curva de pérdida del MLP en el Experimento 5(2) (dos clases, pie derecho).



**Figura 5.9:** Curva de pérdida del MLP en el Experimento 5(3) (tres clases, unificación de lado).



**Figura 5.10:** Curva de pérdida del MLP en el Experimento 5(4) (dos clases, unificación de lado).

En todos los casos, la pérdida en entrenamiento desciende con rapidez, lo que indica que el modelo se adapta con facilidad a los datos de entrenamiento. Sin embargo, esta rapidez de convergencia no se traduce siempre en una buena capacidad de generalización.

### 5.5.2. Métricas y comportamiento por parte

A continuación se resumen los resultados principales de cada configuración:

- **Parte 1 (tres clases, pie derecho):** Se obtuvo una exactitud considerada aceptable a nivel global, pero con desempeño bajo en las clases 1 y 2, lo que indica dificultades del modelo para distinguir las clases minoritarias cuando se consideran las tres categorías simultáneamente.
- **Parte 2 (dos clases, pie derecho):**
  - **Accuracy:** 81.5 %.
  - **F1 ponderado:** 0.76.
  - **Clase 0:** recall perfecto.
  - **Clase 1:** recall muy bajo (16.7 %).

El modelo muestra un buen desempeño global, pero mantiene un sesgo marcado hacia la clase mayoritaria.

- **Parte 3 (tres clases, unificación de lado):**
  - **Accuracy:** 50 %.
  - **F1 macro:** 0.28.

La clase 0 presenta desempeño moderado, mientras que las clases 1 y 2 son prácticamente ignoradas. El incremento en el número total de muestras (200) no se traduce en una mejora proporcional del desempeño.

- **Parte 4 (dos clases, unificación de lado):**

- **Accuracy:** 73.6 %.
- **F1 ponderado:** 0.70.
- **Clase 0:** precisión 0.79, recall 0.90.
- **Clase 1:** recall 0.17.

Aunque el modelo mejora respecto al caso de tres clases con unificación de lado, la sensibilidad hacia la clase minoritaria sigue siendo reducida.

La Tabla 5.4 muestra explícitamente la matriz de confusión para la Parte 4:

Tabla 5.4: Matriz de confusión en test para el Experimento 6(4) (dos clases, unificación de lado).

	Predicho 0	Predicho 1
Real 0	37	4
Real 1	10	2

## 5.6. Resultados del Experimento 6: Embeddings

### 5.6.1. Métricas globales en validación

Con  $n = 40$  muestras de validación (32 No plano / 8 Plano), las métricas consolidadas para los distintos clasificadores se muestran en la Tabla 5.5. Se reportan: *accuracy*, *ROC-AUC*, *recall* por clase y *balanced accuracy*.

De forma complementaria, la Tabla 5.6 resume *accuracy*, *F1* y *balanced accuracy* para los mismos modelos:

Modelo	Accuracy	ROC-AUC	Rec. Plano	Rec. No plano	Balanced Acc.
LogReg	0.825	0.680	0.250	0.969	0.609
LinearSVM	0.800	0.633	0.125	0.969	0.547
kNN	0.825	0.609	0.250	0.969	0.609
ANN (MLP)	0.725	0.699	0.250	0.844	0.547
Random Forest	0.825	<b>0.703</b>	0.250	0.969	0.609

Tabla 5.5: Desempeño en validación sobre *embeddings* Dense-256 (40 imágenes).

Modelo	roc_auc	accuracy	f1	balanced_accuracy
kNN(k=9, dist)	0.6094	0.825	0.3636	0.6094
LogisticRegression	0.6797	0.825	0.3636	0.6094
LinearSVM	0.6328	0.800	0.2000	0.5469
RandomForest	0.7031	0.825	0.3636	0.6094
ANN(MLP)	0.6992	0.725	0.2667	0.5469

Tabla 5.6: Resultados de métricas de distintos modelos sobre *embeddings* Dense-256.

En general, **LogReg**, **kNN** y **Random Forest** alcanzan accuracies en torno a 0.82–0.83, con *balanced accuracy* cercana a 0.61. El Random Forest obtiene el **mejor ROC-AUC** (0.703).

### 5.6.2. Patrones en las matrices de confusión

Las matrices de confusión (no mostradas aquí de forma gráfica) siguen los patrones:

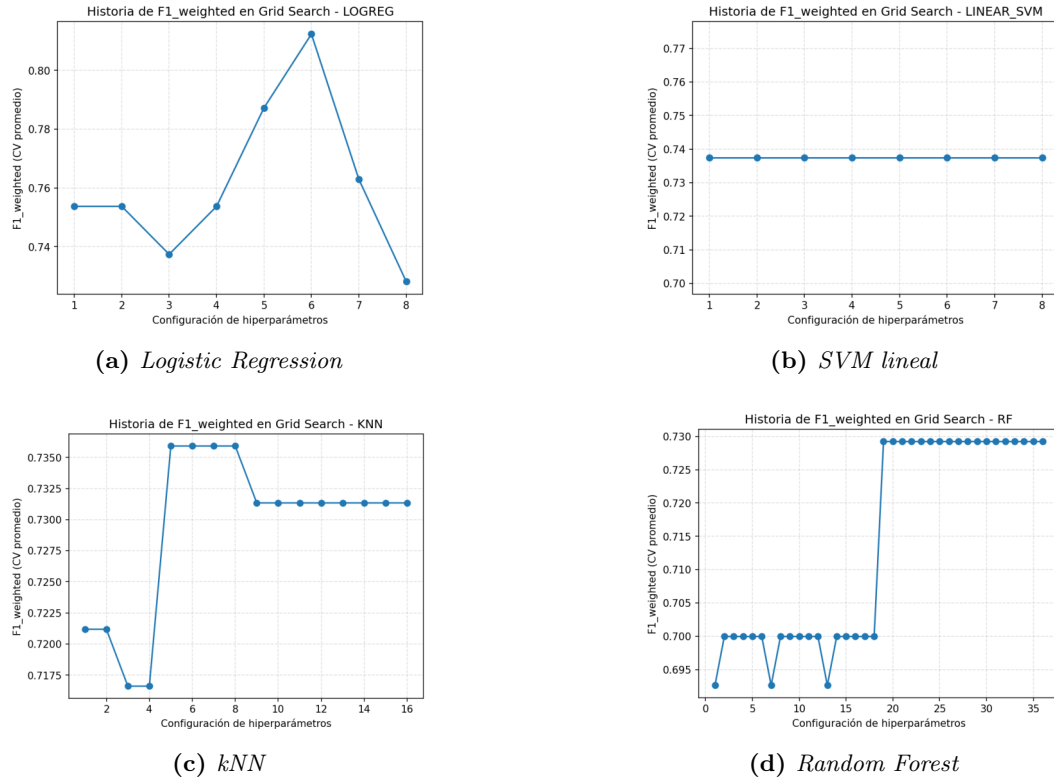
- **LogReg / kNN / Random Forest:** conteos  $\{TN, FP, FN, TP\} = \{31, 1, 6, 2\}$ , lo que implica:
  - **Rec. Plano** = 0.25,
  - **Rec. No plano** = 0.969.
- **LinearSVM:**  $\{TN, FP, FN, TP\} = \{31, 1, 7, 1\} \Rightarrow$  Rec. Plano = 0.125; Rec. No plano = 0.969.
- **ANN:**  $\{TN, FP, FN, TP\} = \{27, 5, 6, 2\} \Rightarrow$  Rec. Plano = 0.25; Rec. No plano = 0.844.

Estos conteos revelan que la mayoría de los modelos son muy conservadores para la clase Plano: priorizan evitar falsos positivos y terminan sacrificando recall en la clase minoritaria.

## 5.7. Resultados del Experimento 7: Matriz de profundidad + embeddings (pie derecho real)

### 5.7.1. Historia de F1\_weighted en la búsqueda de hiperparámetros

Las Figuras 5.11a–5.11d muestran la evolución de la métrica **F1\_weighted** para cada combinación de hiperparámetros evaluada en el Grid Search de los cuatro modelos. El eje X representa el índice de cada configuración dentro de la rejilla, y el eje Y muestra el F1\_weighted promedio obtenido mediante validación cruzada estratificada con 5 particiones.



**Figura 5.11:** Evolución del desempeño en *F1\_weighted* para cada combinación de hiperparámetros evaluada en el Grid Search del Experimento 8. Las curvas permiten identificar la región de hiperparámetros que maximiza el desempeño para (a) Regresión Logística, (b) SVM lineal, (c) *kNN* y (d) *Random Forest*.

A partir de estas trayectorias se seleccionaron, para cada modelo, las configuraciones con mejor *F1\_weighted* promedio en CV, por ejemplo:

- **LogReg:**  $C = 1$ , `class_weight='balanced'`, con  $F1\_weighted(CV) \approx 0.8124$ .
- **SVM lineal:**  $C = 0.01$ , `class_weight=None`, con  $F1\_weighted(CV) \approx 0.7374$ .
- **kNN:**  $k = 5$ , `weights=uniform`,  $p = 1$ , con  $F1\_weighted(CV) \approx 0.7359$ .

- **Random Forest:** `n_estimators = 100, max_depth=None, min_samples_split = 2, class_weight='balanced '`, con `F1_weighted(CV) ≈ 0.7292`.

### 5.7.2. Métricas y matrices de confusión en test

El desempeño final se evaluó sobre el **conjunto de prueba** ( $n = 30$ , con 24 No plano y 6 Plano). La Tabla 5.7 resume las métricas globales por modelo:

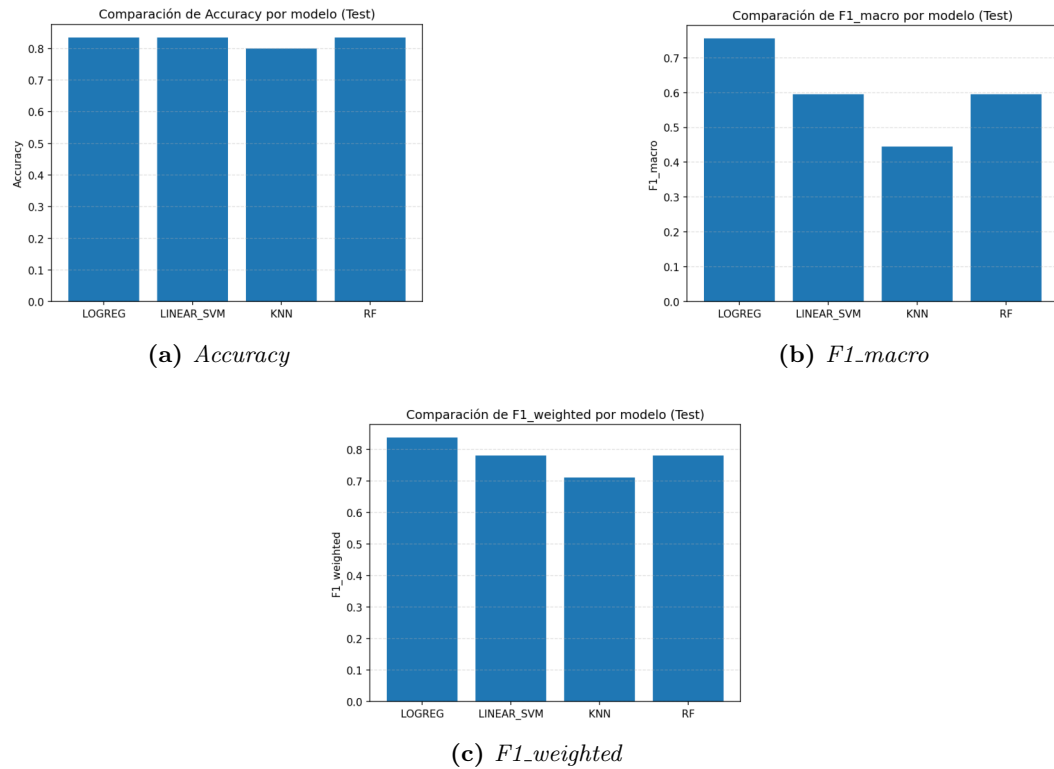
Modelo	Accuracy	F1_macro	F1_weighted
LogReg	0.8333	<b>0.7545</b>	<b>0.8380</b>
LinearSVM	0.8333	0.5957	0.7817
kNN	0.8000	0.4444	0.7111
Random Forest	0.8333	0.5957	0.7817

Tabla 5.7: Desempeño en test ( $n = 30$ ) para el Experimento 8 sobre la representación combinada matriz+embeddings.

Las matrices de confusión permiten analizar el comportamiento por clase:

- **LogReg:**  $\{TN, FP, FN, TP\} = \{21, 3, 2, 4\}$ . Rec. No plano =  $21/24 \approx 0.875$ , rec. Plano =  $4/6 \approx 0.667$ . Es el modelo que logra un **compromiso más equilibrado** entre ambas clases.
- **LinearSVM:**  $\{24, 0, 5, 1\}$ . Rec. No plano = 1.0, rec. Plano =  $1/6 \approx 0.167$ . Se sesga fuertemente hacia la clase No plano.
- **kNN:**  $\{24, 0, 6, 0\}$ . Rec. No plano = 1.0, rec. Plano = 0. Pese a una accuracy de 0.80, **no predice ningún** pie Plano en test.
- **Random Forest:**  $\{24, 0, 5, 1\}$ . Patrón similar a LinearSVM: rec. No plano = 1.0, rec. Plano = 0.167.

La Figura 5.12 ilustra la comparación entre modelos en términos de Accuracy, F1\_macro y F1\_weighted:



**Figura 5.12:** Comparación de métricas del Experimento 7 para los cuatro clasificadores evaluados. (a) Accuracy y (b) F1\_macro se reportan como referencia, dado el desbalance de clases. (c) F1\_weighted, métrica principal del experimento, muestra que la Regresión Logística obtiene el mejor desempeño global sobre el conjunto de prueba.

## 5.8. Resultados del Experimento 8: Embeddings con balanceo mediante SMOTE

### 5.8.1. Desempeño en validación (GridSearchCV)

A partir de la búsqueda en rejilla con validación cruzada estratificada (5 folds) y métrica  $F1_{\text{weighted}}$ , se obtuvieron las siguientes configuraciones óptimas y valores promedio de  $F1_{\text{weighted}}$  en CV para cada modelo:

- **LOGREG:**

$C = 1,$   
`class_weight = None,`  
 $F1_{\text{weighted}}(\text{CV}) \approx 0.8606.$

- **LINEAR\_SVM:**

$C = 0.1,$   
`class_weight = None,`  
 $F1_{\text{weighted}}(\text{CV}) \approx 0.8606.$

- **KNN:**

$k = 5,$   
`weights = 'uniform',`  
 $F1_{\text{weighted}}(\text{CV}) \approx 0.7290.$

- **RF:**

$n_{\text{estimators}} = 100,$   
`max_depth = None,`  
`min_samples_split = 2,`  
 $F1_{\text{weighted}}(\text{CV}) \approx 0.8606.$

Estos valores reflejan que, en validación cruzada, LOGREG, LINEAR\_SVM y RF obtienen desempeños muy similares y altos en términos de F1\_weighted, mientras que KNN queda ligeramente por debajo.

### 5.8.2. Métricas y matrices de confusión en prueba

Tras seleccionar los mejores hiperparámetros para cada clasificador, se evaluó el modelo resultante sobre el conjunto de prueba ( $n = 30$ , con 24 No plano y 6 Plano). La Tabla 5.8 resume las métricas globales:

Tabla 5.8: Desempeño en test sobre *embeddings* Dense-256 de pie derecho con balanceo SMOTE (30 imágenes).

Modelo	Accuracy	F1_macro	F1_weighted
LOGREG	0.967	0.944	0.965
LINEAR_SVM	0.867	0.760	0.856
KNN	0.800	0.744	0.816
RF	0.900	0.853	0.903

Las matrices de confusión asociadas permiten analizar el comportamiento por clase:

- **LOGREG:**

$$\begin{bmatrix} 24 & 0 \\ 1 & 5 \end{bmatrix} \Rightarrow \text{Rec. No plano} = 1.00, \quad \text{Rec. Plano} = 5/6 \approx 0.83.$$

- **LINEAR\_SVM:**

$$\begin{bmatrix} 23 & 1 \\ 3 & 3 \end{bmatrix} \Rightarrow \text{Rec. No plano} \approx 0.96, \quad \text{Rec. Plano} = 3/6 = 0.50.$$

■ **KNN:**

$$\begin{bmatrix} 19 & 5 \\ 1 & 5 \end{bmatrix} \Rightarrow \text{Rec. No plano} \approx 0.79, \quad \text{Rec. Plano} = 5/6 \approx 0.83.$$

■ **RF:**

$$\begin{bmatrix} 22 & 2 \\ 1 & 5 \end{bmatrix} \Rightarrow \text{Rec. No plano} \approx 0.92, \quad \text{Rec. Plano} = 5/6 \approx 0.83.$$

En conjunto, LOGREG y RF logran altos valores de **F1\_weighted** y **recall de Plano**, mientras que LINEAR\_SVM y KNN presentan compromisos distintos entre sensibilidad a la clase positiva y reconocimiento de la clase No plano.

---

## CAPÍTULO 6

---

# Discusión

A lo largo de esta investigación se llevaron a cabo ocho experimentos que, aún cuando emplearon representaciones y modelos de distinta naturaleza, conformaron un proceso progresivo de refinamiento metodológico. La secuencia experimental permitió transitar desde enfoques basados en una sola característica morfométrica hasta representaciones latentes de mayor riqueza estructural, siempre con el propósito de comprender qué aspectos de la información plantar resultaban verdaderamente útiles para la detección del pie plano.

El primer conjunto de experimentos exploró las limitaciones de trabajar únicamente con el *Factor\_Standard*. La estimación de densidades mediante KDE y el uso de un clasificador Bayesiano evidenciaron que una variable aislada no es suficiente para describir la complejidad morfológica del arco plantar. La fuerte superposición entre distribuciones y el desbalance en las clases provocaron que el modelo colapsara hacia la clase mayoritaria, generando métricas aparentemente aceptables pero sin capacidad real para identificar casos de pie plano. Incluso al aplicar un balanceo por truncado, la capacidad discriminativa continuó siendo reducida; el problema no era el balance del conjunto, sino la insuficiencia de la representación. El caso binario confirmó esta observación, pues aún con menos clases, el solapamiento persistió y el modelo siguió ignorando la clase minoritaria. Esta primera etapa dejó claro que era necesario incorporar descriptores con mayor contenido geométrico y espacial.

El segundo bloque consistió en analizar las matrices de profundidad completas del pie. Este enfoque, aunque más rico en principio, introdujo un escenario típico de *alta dimensionalidad frente a un número reducido de muestras*. El MLP utilizado como clasificador tendió al sobreajuste desde etapas tempranas, mostrando exactitu-

des elevadas en entrenamiento pero desempeño limitado en validación y prueba. Las clases minoritarias continuaron sin ser correctamente identificadas, y ni el aumento sintético mediante espejo ni las variaciones de particionado modificaron sustancialmente este comportamiento. La lateralidad del pie tampoco mostró un efecto relevante en los resultados. El análisis conjunto de estos experimentos dejó en evidencia que, aunque la matriz de profundidad contiene información valiosa, su dimensionalidad extrema impide que los modelos capturen patrones de forma eficiente. Era necesario encontrar una representación más compacta y estructurada.

La tercera etapa del estudio introdujo representaciones latentes obtenidas mediante una CNN preentrenada. Estos *embeddings* demostraron ser significativamente más informativos que las matrices de profundidad crudas y, al mismo tiempo, mucho más manejables desde el punto de vista computacional. Los modelos clásicos entrenados con estas representaciones lograron mejorar de manera notable su estabilidad y su capacidad para distinguir entre clases. Sin embargo, el desbalance continuó afectando la detección del pie plano, lo que motivó la aplicación de SMOTE. El efecto del balanceo fue particularmente evidente cuando se aplicó sobre el espacio latente: los modelos no sólo incrementaron su sensibilidad hacia la clase minoritaria, sino que lo hicieron sin sacrificar el desempeño global. La coherencia geométrica entre muestras permitió que las interpolaciones de SMOTE produjeran ejemplos sintéticos plausibles, lo que repercutió directamente en un aumento del recall de la clase Plano y en métricas generales más elevadas. Esta última etapa consolidó la idea de que la representación es tan importante como el modelo utilizado.

La investigación también enfrentó limitaciones relacionadas con la recolección de datos. El proyecto contemplaba inicialmente la construcción de una base de datos infantil; sin embargo, los permisos requeridos para realizar capturas en instituciones públicas no fueron autorizados dentro del periodo disponible. Aunque se buscó obtener muestras en centros deportivos y recreativos, la participación fue baja y la cantidad de datos insuficiente para realizar un análisis significativo. Por ello, la tesis se centró en la base de datos de adultos. Esta situación no invalida los resultados obtenidos, pero sí delimita el alcance del estudio y señala la importancia de continuar con la recolección de datos infantiles en trabajos futuros.

En conjunto, los experimentos desarrollados permiten comprender de manera más completa los factores que intervienen en la clasificación del pie plano a partir de imágenes de profundidad. Se observó que la representación es un componente crítico: una variable aislada resulta insuficiente, la matriz cruda plantea retos de dimensionalidad que afectan la estabilidad de los modelos, y las representaciones latentes moderadas permiten extraer información de manera más consistente. Asimismo, quedó claro que el desbalance en las clases influye de manera decisiva en la detección de la clase minoritaria, por lo que técnicas como SMOTE son indispensables cuando se trabaja con conjuntos reducidos.

Los resultados muestran que el análisis basado en *embeddings* y estrategias de balanceo constituye la vía más sólida para avanzar hacia sistemas de apoyo al diagnóstico del pie plano. La coherencia interna de los hallazgos y la secuencia lógica de los experimentos permiten trazar una línea clara de evolución metodológica, desde enfoques sencillos pero limitados, hasta representaciones capaces de capturar la complejidad geométrica del arco plantar. Este panorama general establece bases firmes para la construcción de herramientas clínicas asistidas por visión por computadora y abre múltiples líneas de investigación destinadas a robustecer los métodos desarrollados en este trabajo.

---

## CAPÍTULO 7

---

# Conclusiones

En este trabajo se desarrolló una metodología para el análisis de huellas plantares utilizando imágenes de profundidad y técnicas de aprendizaje automático, con el propósito de apoyar la detección preventiva del pie plano. A lo largo de la investigación se evaluaron diferentes representaciones de la información y distintos modelos de clasificación, con el fin de identificar cuáles de ellos eran capaces de capturar de manera más fiel las variaciones morfológicas del arco plantar.

Las primeras etapas del estudio permitieron constatar que es posible extraer información morfológica confiable a partir de mapas de profundidad corregidos. Métricas como el área proyectada, el volumen bajo la curva y la altura promedio mostraron comportamientos reproducibles y coherentes con la estructura anatómica del pie, aunque su capacidad de discriminación entre clases fue limitada cuando se utilizaron de forma aislada. Este hallazgo subraya la necesidad de representaciones que integren mejor la geometría tridimensional del arco.

La comparación con los índices clínicos tradicionales reveló una correspondencia importante entre ambos enfoques. Tanto el índice de Staheli como el índice de Chipaux–Smirak mantuvieron patrones consistentes con las mediciones derivadas de la cámara, lo cual sugiere que las tecnologías basadas en profundidad pueden complementar adecuadamente los métodos utilizados actualmente en la práctica clínica. Esta relación refuerza la perspectiva de que un sistema automatizado de bajo costo puede fungir como apoyo para la valoración inicial del pie plano.

Los análisis posteriores evidenciaron que la matriz de profundidad completa presenta retos significativos debido a su elevada dimensionalidad. Los modelos entrena-

dos con estas matrices tendieron al sobreajuste y mostraron poca sensibilidad hacia la clase correspondiente al pie plano. En contraste, los *embeddings* generados mediante una CNN preentrenada demostraron ser representaciones más estables y discriminativas. Su menor dimensionalidad, combinada con su capacidad para capturar rasgos geométricos relevantes, permitió a clasificadores clásicos como la Regresión Logística, KNN y Random Forest mejorar notablemente su desempeño.

El uso de SMOTE sobre el espacio latente marcó un punto clave en el desarrollo del proyecto. Al operar en un espacio de 256 dimensiones, las interpolaciones generadas por SMOTE mantuvieron coherencia geométrica y favorecieron un incremento sustancial en la sensibilidad hacia la clase minoritaria. Modelos entrenados bajo este esquema alcanzaron niveles altos de exactitud global y recuperaron la clase Plano con valores de recall cercanos al 0.83, lo que constituye un avance notable considerando el desbalance natural del conjunto de datos.

En términos generales, los resultados permiten afirmar que las cámaras de profundidad ofrecen una alternativa accesible y no invasiva para caracterizar la morfología del arco plantar; que la representación es un factor decisivo para el desempeño de los modelos, siendo los *embeddings* una opción particularmente eficiente; y que el tratamiento explícito del desbalance de clases es indispensable para recuperar información relevante sobre el pie plano. La integración de elementos clínicos tradicionales con técnicas modernas de visión por computadora abre, además, la posibilidad de diseñar herramientas de apoyo diagnóstico que sean económicas, reproducibles y fáciles de implementar en entornos no especializados.

El significado de este trabajo radica en mostrar que es posible construir un sistema funcional de análisis plantar con infraestructura accesible, capaz de generar mediciones objetivas que complementen la evaluación clínica. Si bien los resultados no sustituyen la opinión de un especialista, sí proporcionan un mecanismo cuantitativo que puede servir para tamizajes iniciales, estudios poblacionales o seguimiento de la evolución del arco en etapas tempranas de desarrollo.

A partir de las limitaciones encontradas se identifican varias líneas de trabajo futuro. Entre ellas, resulta fundamental ampliar la base de datos, en especial en población infantil, con el fin de estudiar la evolución del arco plantar y evaluar la capacidad de generalización del sistema. Asimismo, es importante considerar particionados por individuo para medir el desempeño frente a sujetos no vistos durante el entrenamiento. La exploración de modelos entrenados de manera *end-to-end* sobre datos RGB-D, la integración de mapas de presión y el diseño de evaluaciones clínicas más amplias representan oportunidades claras para extender y fortalecer el enfoque desarrollado en esta tesis.

En conjunto, los resultados obtenidos demuestran que el análisis de profundidad, reforzado mediante representaciones latentes y técnicas de balanceo, constituye un enfoque prometedor para el análisis automatizado de huellas plantares. Este trabajo sienta bases metodológicas y experimentales que pueden servir como punto de partida para desarrollos más robustos, así como para la eventual implementación de sistemas clínicos de apoyo destinados a la detección temprana de alteraciones del arco plantar.

---

## Bibliografía

- [1] R. L. Drake, W. Vogl, and A. W. M. Mitchell, *Gray's Anatomy for Students*, 3rd ed. Philadelphia, PA: Elsevier Health Sciences, 2015.
- [2] L. Marino, Y. González, C. González, and M. Rupcich, "Propuesta de modelo multisegmento del pie para el análisis de marcha," *Revista de la Facultad de Ingeniería Universidad Central de Venezuela*, vol. 28, no. 4, pp. 143–158, 2013, recuperado en 11 de octubre de 2024. [Online]. Available: [http://ve.scielo.org/scielo.php?script=sci\\_arttext&pid=S0798-40652013000400015&lng=es&tlng=es](http://ve.scielo.org/scielo.php?script=sci_arttext&pid=S0798-40652013000400015&lng=es&tlng=es)
- [3] R. Cailliet, *Anatomía funcional: biomecánica*. Madrid, España: Editorial Marbán, 2001.
- [4] S. Raschka, *Python Machine Learning: Unlock deeper insights into machine learning with this vital guide to cutting-edge predictive analytics*, 1st ed. Livery Place, 35 Livery Street, Birmingham B3 2PB, UK: Packt Publishing Ltd., September 2015.
- [5] U. Kanatli, H. Yetkin, and E. Cila, "Footprint and radiographic analysis of the feet," *Journal of Pediatric Orthopedics*, vol. 21, no. 2, pp. 225–228, Mar–Apr 2001.
- [6] United Nations Children's Fund (UNICEF), "Case study: Mexico," Panama, Republic of Panama, July 2022. [Online]. Available: <https://www.unicef.org/lac/media/36911/file/Case%20study:%20Mexico.pdf>
- [7] N. Shibuya, D. Jupiter, L. Ciliberti, V. VanBuren, and J. Fontaine, "Characteristics of adult flatfoot in the united states," *Journal of Foot and Ankle Surgery*, vol. 49, 2010.
- [8] Gobierno de México, "Detecta a tiempo el pie plano," n.d., acceso el 22 de enero de 2025. [Online]. Available: <https://www.gob.mx/salud/articulos/detec>

ta-a-tiempo-el-pie-plano#:~:text=En%20M%C3%A9xico%20entre%2015%20y,%20rodilla%20cadera%20y%20columna.

- [9] K. A. I., *Fisiología Articular*. Panamericana, 2012.
- [10] S. Standring, Ed., *Gray's Anatomy: The anatomical basis of clinical practice*, 41st ed. Elsevier, 2016.
- [11] K. L. Moore, A. M. R. Agur, and A. F. Dalley, *Essential Clinical Anatomy*, 5th ed. Philadelphia, PA: Lippincott Williams & Wilkins, 2014.
- [12] DeGowin, *Examen diagnóstico*, 11th ed. Estados Unidos: McGraw Hill Education Inc., 2021, pagina de definicion de pie plano. Traducción gestionada por M&N Medical Solutrad S.A. de C.V.
- [13] T. Kobayashi, K. Hirota, R. Otsuki, J. Onodera, T. Kodesho, and K. Taniguchi, "Morphological and mechanical characteristics of the intrinsic and extrinsic foot muscles under loading in individuals with flat feet," *Gait Posture*, vol. 108, pp. 15–21, Feb. 2024.
- [14] T. Suciati, M. Rulan Adnindya, I. Seta Septadina, and P. Putri Pratiwi, "Correlation between flat feet and body mass index in primary school students," *J. Phys. Conf. Ser.*, vol. 1246, no. 1, p. 012063, Jul. 2019.
- [15] H. B. Skinner and P. J. McMahon, "Capítulo 8: Diagnóstico y tratamiento en ortopedia," in *Diagnóstico y tratamiento en ortopedia*, 5th ed., H. B. Skinner and P. J. McMahon, Eds. Estados Unidos: McGraw Hill Education, 2021.
- [16] C. Kerr, A. Zavatsky, T. Theologis, and J. Stebbins, "Kinematic differences between neutral and flat feet with and without symptoms as measured by the oxford foot model," *Gait & Posture*, vol. 67, pp. 213–218, 2019.
- [17] Instituto Mexicano del Seguro Social (IMSS), *Abordaje Diagnóstico del Pie Plano en Niñas/ Niños y las/los Adolescentes en el Primer Nivel de Atención*, Instituto Mexicano del Seguro Social, México, D.F., 2015, guía de Práctica Clínica, IMSS-779-15. [Online]. Available: <http://imss.gob.mx/profesionales-salud/gpc>

- [18] D. A. Neumann, *Kinesiology of the Musculoskeletal System: Foundations for Rehabilitation*, 3rd ed. Elsevier, 2017.
- [19] U. Kanatli, H. Yetkin, and E. Cila, “Footprint and radiographic analysis of the feet,” *Journal of Pediatric Orthopaedics*, vol. 21, no. 2, pp. 225–228, 2001. [Online]. Available: <https://doi.org/10.1097/01241398-200103000-00005>
- [20] K. Inui, K. Ikoma, K. Imai, S. Ohashi, M. Maki, M. Kido, Y. Hara, Y. Oka, H. Fujiwara, and T. Kubo, “Examination of the correlation between foot morphology measurements using pedography and radiographic measurements,” *Journal of Foot and Ankle Surgery*, vol. 56, no. 1, pp. 61–65, 2017.
- [21] H. A. Banwell, A. Paris, A. Mackintosh, and S. Williams, “Paediatric flexible flat foot: how are we measuring it and are we getting it right? a systematic review,” *Journal of Foot and Ankle Research*, vol. 11, no. 21, 2018. [Online]. Available: <https://doi.org/10.1186/s13047-018-0264-3>
- [22] S. Alsancak, S. Guner, E. Güven, N. Yalcin, S. Karahan, T. Türker, and T. Aydin, “Paediatric flat foot and foot dimension in central anatolia,” *BMC Pediatrics*, vol. 21, no. 200, pp. 1–9, 2021. [Online]. Available: <https://doi.org/10.1186/s12887-021-02645-9>
- [23] C. Gonzalez-Martin, S. Pita-Fernandez, T. Seoane-Pillado, B. Lopez-Calviño, S. Pertega-Diaz, and V. Gil-Guillen, “Variability between clarke’s angle and chippaux-smirak index for the diagnosis of flat feet,” *Colomb Med (Cali)*, vol. 48, no. 1, pp. 25–31, 2017. [Online]. Available: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5438225/>
- [24] J. A. Ramírez-Bautista, A. Hernández-Zavala, S. L. Chaparro-Cárdenas, and J. A. Huerta-Ruelas, “Revisión sobre el análisis de datos plantares para el diagnóstico de enfermedades,” *Biocybernetics and Biomedical Engineering*, vol. 38, no. 2, pp. 342–361, 2018.
- [25] W. R. Ledoux and H. J. Hillstrom, “The distributed plantar vertical force of neutrally aligned and pes planus feet,” *Gait & Posture*, vol. 15, no. 1, pp. 1–9, 2002. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0966636201001655>

- [26] L. Vorlickova and P. Korvas, "Evaluation of rehabilitation influence on flat foot in children by plantar pressure analysis," *Journal of Human Sport and Exercise*, vol. 9, no. 1, pp. S526–S532, 2014.
- [27] J. M. C. Rodríguez, "Implementación y elaboración de dispositivo electrónico para el análisis de huella plantar mediante obtención y comparación de imágenes," Tesis Profesional, Universidad de Guanajuato, Campus Irapuato – Salamanca, División de Ingenierías, Salamanca, Gto., agosto 2017.
- [28] J. J. Rivas Jiménez, "Sistema de análisis de huella plantar para su diagnóstico en tiempo real, implementado en tecnología raspberry pi," Master's thesis, Universidad de Guanajuato, Campus Irapuato-Salamanca, División de Ingenierías, Salamanca, Guanajuato, México, Mayo 2018, tesis de Maestría en Ingeniería Eléctrica.
- [29] W. Sawangphol, P. Panphattarasap, P. Praiwattana, J. Krajangka, T. Noraset, and D. Prommin, "Foot arch classification via ml-based image classification," *Computer-Aided Design and Applications*, vol. 20, no. 4, pp. 600–613, 2023.
- [30] J. A. Ramirez-Bautista, J. A. Huerta-Ruelas, L. T. Kóczy, M. F. Hatwágner, S. L. Chaparro-Cárdenas, and A. Hernández-Zavala, "Classification of plantar foot alterations by fuzzy cognitive maps against multi-layer perceptron neural network," *Biocybernetics and Biomedical Engineering*, vol. 40, no. 1, pp. 404–414, 2020. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0208521620300048>
- [31] K. W. Leung, "Prevalence and associated factors of flat foot among 6 to 16 aged chinese children and adolescents: A cross-sectional study," M.Phil. thesis, The Hong Kong Polytechnic University, Hong Kong, 2023, adviser: Ming Zhang; 89 pages, includes color illustrations. [Online]. Available: <https://theses.lib.polyu.edu.hk/handle/200/12380>
- [32] Y. Hirose, S. Koyama, M. Ishii, S. Saitou, M. Takemoto, Y. Nose, A. Inoue, Y. Sakata, Y. Sugiura, T. Kabe, M. Usuda, S. Kasuga, M. Mori, A. Odagawa, and T. Tanaka, "A 250 m direct time-of-flight range image sensor based on a sub-range image synthesizing technique and a cmos image sensor with

- a vertical avalanche photodiode,” *Sensors*, vol. 18, no. 11, p. 3642, 2018. [Online]. Available: <https://www.mdpi.com/1424-8220/18/11/3642>
- [33] L. Fu, F. Gao, J. Wu, R. Li, M. Karkee, and Q. Zhang, “Application of consumer rgb-d cameras for fruit detection and localization in field: A critical review,” *Computers and Electronics in Agriculture*, vol. 177, p. 105687, 2020. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0168169920319530>
- [34] F. Liu, D. Chen, J. Zhou, and F. Xu, “A review of driver fatigue detection and its advances on the use of rgb-d camera and deep learning,” *Engineering Applications of Artificial Intelligence*, vol. 116, p. 105399, 2022. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0952197622002169>
- [35] W. Yanmin and W. Shuangfeng, *Point Cloud Management Based on Depth Images*. China: Surveying and Mapping Press, 2013, chinese Edition.
- [36] P. M. B. S. Dias, V. Sequeira, F. Vaz, and J. G. M. Gonçalves, “Combining intensity and range images for 3d modelling,” in *Proceedings of the 2003 International Conference on Image Processing (ICIP)*. IEEE, 2003, pp. 593–596.
- [37] F. Remondino and S. El-Hakim, “Image-based 3d modelling: A review,” *The Photogrammetric Record*, vol. 21, no. 115, pp. 269–291, 2006. [Online]. Available: <https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1477-9730.2006.00383.x>
- [38] L. Fu, F. Gao, J. Wu, R. Li, M. Karkee, and Q. Zhang, “Application of consumer rgb-d cameras for fruit detection and localization in field: A critical review,” *Computers and Electronics in Agriculture*, vol. 177, p. 105687, 2020. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0168169920319530>
- [39] Intel RealSense. (2019, 7) Beginner’s guide to depth (updated). Accedido el 5 de junio de 2025. [Online]. Available: <https://www.intelrealsense.com/beginner-rs-guide-to-depth/>

- [40] Carnegie Mellon Computational Imaging, “Computational imaging lecture,” <https://www.youtube.com/watch?v=uco5UlfPheQ>, 12 2024, accedido el 5 de junio de 2025.
- [41] A. Wunsch and K. Matuschka, “A controlled trial to determine the efficacy of red and near-infrared light treatment in patient satisfaction, reduction of fine lines, wrinkles, skin roughness, and intradermal collagen density increase,” *Photomedicine and Laser Surgery*, vol. 32, no. 2, pp. 93–100, 2014. [Online]. Available: <https://pubmed.ncbi.nlm.nih.gov/24286286/>
- [42] Intel RealSense Developer Documentation, “Stereo depth cameras for phones,” 2024, accessed: 2025-06-10. [Online]. Available: <https://dev.intelrealsense.com/docs/stereo-depth-cameras-for-phones>
- [43] R. Szeliski, *Depth Estimation*. Cham: Springer International Publishing, 2022, pp. 595–638. [Online]. Available: [https://doi.org/10.1007/978-3-030-34372-9\\_12](https://doi.org/10.1007/978-3-030-34372-9_12)
- [44] P. I. Corke, *Robotics, Vision and Control: Fundamental Algorithms in MATLAB*, ser. Springer Tracts in Advanced Robotics. Springer-Verlag Berlin Heidelberg, 2011, vol. 73, library of Congress Control Number: 2011934624.
- [45] M. Büyük. (2023, 11) Obtaining point cloud from depth images with intel realsense d-435 camera. Accedido el 9 de junio de 2025. [Online]. Available: <https://medium.com/@mustafaboyuk24/obtaining-point-cloud-from-depth-images-with-intel-realsense-d-435-camera-144e8ef9260d>
- [46] I. Corporation, *Intel® RealSense™ D400 Series Product Family Datasheet*, Oct 2023, document Number: 21345, Incluye especificaciones técnicas de la D435i. [Online]. Available: <https://www.intelrealsense.com/depth-camera-d435i/>
- [47] OmniVision Technologies, “Ov9282: 1-megapixel global shutter image sensor,” <https://www.ovt.com/products/ov9282/>, 2024, accedido el 5 de junio de 2025.
- [48] T. Iglesias, A. Salmon, J. Scholtz, R. Hedegore, J. Borgendale, B. Runnels, and N. McKimpton, *Camera Computer Interfaces*. John Wiley & Sons, Ltd, 2006, ch. 7, pp. 427–509. [Online]. Available: <https://onlinelibrary.wiley.com/doi/abs/10.1002/9783527610136.ch7>

- [49] Intel RealSense, “Stereo depth cameras for phones,” 2024, Último acceso: 13 de junio de 2025. [Online]. Available: <https://dev.intelrealsense.com/docs/stereo-depth-cameras-for-phones>
- [50] L. G. Shapiro and G. C. Stockman, *Computer Vision*. Upper Saddle River, NJ, USA: Prentice Hall, 2001. [Online]. Available: <https://www.amazon.com/Computer-Vision-Linda-G-Shapiro/dp/0130307963>
- [51] F. Garcia-Lamont, J. Cervantes, A. López, and L. Rodriguez, “Segmentation of images by color features: A survey,” *Neurocomputing*, vol. 292, pp. 1–27, May 2018. [Online]. Available: <https://doi.org/10.1016/j.neucom.2018.01.091>
- [52] M. Sonka, V. Hlaváč, and R. Boyle, *Image Processing, Analysis, and Machine Vision*, 4th ed. Boston, MA: Cengage Learning, 2014, incluye acceso a materiales complementarios online. [Online]. Available: <https://www.cengage.com/c/image-processing-analysis-and-machine-vision-4e-sonka/9781133593607/>
- [53] P. M. Pavlopoulos, S. Zimeras, N. Kavantzias, P. Korkolopoulou, E. Agapitos, and E. Patsouris, “Segmentation of transitional cell carcinoma nuclei by non-supervised thresholding in different color spaces,” *ANALYTICAL AND QUANTITATIVE CYTOLOGY AND HISTOLOGY*, vol. 29, no. 4, pp. 271–278, AUG 2007.
- [54] S. Yoon and J. Min, “An intelligent automatic early detection system of forest fire smoke signatures using gaussian mixture model,” *Journal of Information Processing Systems*, vol. 9, no. 4, pp. 621–632, 2013, indexed in Scopus and KCI.
- [55] S. Spurgeon, J. A. Benediktsson, A. Bose, J. Duncan, A. Moeness, D. S. Naidu, B. Razavi, J. Lyke, H. Li, B. Johnson, J. Reed, D. Spinellis, A. Drobot, T. Robertazzi, and A. M. Tekalp, Eds., *Demystifying Deep Learning*, ser. IEEE Press Series on Technology Management, Innovation and Leadership. 445 Hoes Lane, Piscataway, NJ 08854, USA: IEEE Press, 2023, part of the IEEE Press Series on Emerging Technologies. [Online]. Available: <https://ieeexplore.ieee.org/book/9666097>

- [56] E. F. Caicedo Bravo and J. A. López Sotelo, *Una aproximación práctica a las redes neuronales artificiales*, primera reimpresión ed., ser. Colección Ingeniería. Cali, Colombia: Programa Editorial, Universidad del Valle, 2020, primera Edición Impresa. [Online]. Available: <https://programaeditorial.univalle.edu.co>
- [57] S. J. Pan and Q. Yang, “A survey on transfer learning,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, p. 1345, October 2010.
- [58] R. Ribani and M. Marengoni, “A survey of transfer learning for convolutional neural networks,” in *2019 32nd SIBGRAPI Conference on Graphics, Patterns and Images Tutoriais (SIBGRAPI-T)*. São Paulo, SP, Brazil: IEEE, October 2019, pp. 01 302–000, programa de Pós-Graduação em Engenharia Elétrica e Computação, Universidade Presbiteriana Mackenzie. [Online]. Available: <https://doi.org/10.1109/SIBGRAPI-T.2019.00010>
- [59] D. F. K. Gom-os, “Fruit classification using colorized depth images,” *International Journal of Advanced Computer Science and Applications*, vol. 14, no. 5, pp. 1025–1031, 2023. [Online]. Available: <https://ijacsa.thesai.org/>
- [60] Intel Corporation, “Align depth to color - python example,” <https://github.com/IntelRealSense/librealsense/blob/master/wrappers/python/examples/align-depth2color.py>, 2024, accedido: julio 2025.
- [61] —, “Align depth to color - python example,” <https://github.com/IntelRealSense/librealsense/blob/master/wrappers/python/examples/align-depth2color.py>, 2024, Último acceso: julio 2025.