



UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE GUAYAQUIL
CARRERA DE MECATRÓNICA

**DESARROLLO DE UN SISTEMA DE ANÁLISIS DE IMÁGENES DE
RAYOS X PARA LA DETECCIÓN DE AFECCIONES DENTALES
UTILIZANDO VISIÓN ARTIFICIAL**

Trabajo de titulación previo a la obtención del
Título de Ingeniero en Mecatrónica

AUTOR: Claudia Angélica Franco Herrera
TUTOR: Ing. Alberto Santiago Ramírez Farfán

Guayaquil - Ecuador
2024

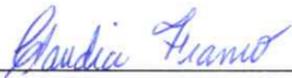
CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA DEL TRABAJO DE TITULACIÓN

Yo, **Claudia Angélica Franco Herrera** con documento de identificación N° **0956186019** manifiesto que:

Soy autora y responsable del presente trabajo; y, autorizo a que sin fines de lucro la Universidad Politécnica Salesiana pueda usar, difundir, reproducir o publicar de manera total o parcial el presente trabajo.

Guayaquil, 13 de Septiembre del año 2024

Atentamente,



Claudia Angélica Franco Herrera
0956186019

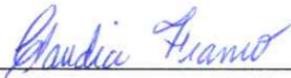
**CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE TITULACIÓN A LA
UNIVERSIDAD POLITÉCNICA SALESIANA**

Yo, **Claudia Angélica Franco Herrera** con documento de identificación N° **0956185019** expreso mi voluntad y por medio del presente documento cedo a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que soy autora del **Dispositivo Tecnológico: DESARROLLO DE UN SISTEMA DE ANÁLISIS DE IMÁGENES DE RAYOS X PARA LA DETECCIÓN DE AFECCIONES DENTALES UTILIZANDO VISIÓN ARTIFICIAL**, el cual ha sido desarrollado para optar por el título de: Ingeniero en Mecatrónica, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En concordancia con lo manifestado, suscribo este documento en el momento que hacemos la entrega del trabajo final en formato digital a la Biblioteca de la Universidad Politécnica Salesiana.

Guayaquil, 13 de Septiembre del año 2024

Atentamente,



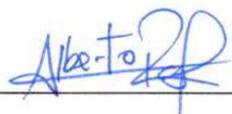
Claudia Angélica Franco Herrera
0956186019

CERTIFICADO DE DIRECCIÓN DEL TRABAJO DE TITULACIÓN

Yo, **Alberto Santiago Ramírez Farfán**, docente de la Universidad Politécnica Salesiana, declaro que bajo mi tutoría fue desarrollado el trabajo de titulación: **DESARROLLO DE UN SISTEMA DE ANÁLISIS DE IMÁGENES DE RAYOS X PARA LA DETECCIÓN DE AFECCIONES DENTALES UTILIZANDO VISIÓN ARTIFICIAL**, realizado por **Claudia Angélica Franco Herrera** con documento de identificación N° **0956186019** obteniendo como resultado final el trabajo de titulación bajo la opción **Dispositivo Tecnológico** que cumple con todos los requisitos determinados por la Universidad Politécnica Salesiana.

Guayaquil, 13 de Septiembre del año 2024

Atentamente,



Ing. Alberto Santiago Ramírez Farfán, Mg.
0923348890

DEDICATORIA

Este trabajo de titulación está dedicado a Dios, a mis padres Patricia Herrera y Ángel Franco, mis hermanos, Roberto, Emily y Rocío, mis ángeles en el cielo, Esteban, Lola y William y mi familia materna, ya que han sido mi motor para alcanzar este importante objetivo en mi vida. Me lo dedico a mí, por todo los sacrificios y adversidades que tuve que superar a lo largo de este tiempo para poder llegar a este punto.

Agradezco así mismo a mi familia y a los docentes que a lo largo de mi carrera me brindaron sus conocimientos y apoyo.

Claudia Angélica Franco Herrera

RESUMEN

Este proyecto presenta el desarrollo de un sistema automatizado de análisis de imágenes de rayos X dentales, utilizando redes neuronales convolucionales (CNN) para la detección de afecciones dentales, específicamente caries y lesiones periapicales. La propuesta está orientada a mejorar la precisión en los diagnósticos odontológicos mediante el uso de técnicas avanzadas de visión artificial.

El sistema se apoya en un conjunto de datos de radiografías dentales anotadas y emplea un enfoque supervisado de aprendizaje automático para entrenar la CNN. Se utilizó PyTorch como framework de desarrollo, Jupyter Notebook para la escritura y ejecución del código, y Anaconda para la gestión del entorno y las dependencias. A lo largo del desarrollo, se evaluó el modelo de detección de objetos Faster R-CNN, considerado por su alta precisión en el diagnóstico de afecciones dentales. Además, se aplicaron técnicas de Data Augmentation para mejorar la diversidad del conjunto de datos y aumentar la robustez del modelo.

El trabajo incluye técnicas de preprocesamiento de imágenes y anotación manual de datos mediante herramientas especializadas como LabelMe.

Finalmente, se plantea el uso de este sistema en el ámbito clínico, donde podría asistir a odontólogos en la identificación rápida y precisa de afecciones, optimizando el tiempo de diagnóstico y reduciendo la dependencia de la experiencia visual del especialista.

Palabras claves: sistemas de visión artificial, redes neuronales convolucionales, diagnóstico odontológico, Faster R-CNN, PyTorch, Anaconda, Jupyter Notebook, Data Augmentation, caries, lesiones periapicales.

ABSTRACT

This project presents the development of an automated system for the analysis of dental X-ray images using convolutional neural networks (CNN) for the detection of dental conditions, specifically caries and periapical lesions. The proposal is aimed at improving the accuracy of dental diagnoses through the use of advanced computer vision techniques.

The system is based on a dataset of annotated dental X-rays and employs a supervised machine learning approach to train the CNN. PyTorch was used as the development framework, Jupyter Notebook for writing and running the code, and Anaconda for environment and dependency management. Throughout the development, the Faster R-CNN object detection model was evaluated, recognized for its high accuracy in diagnosing dental conditions. Additionally, Data Augmentation techniques were applied to enhance the diversity of the dataset and increase the robustness of the model.

The work includes image preprocessing techniques and manual data annotation using specialized tools like LabelMe.

Finally, the use of this system is proposed in the clinical field, where it could assist dentists in the quick and accurate identification of conditions, optimizing diagnosis time and reducing dependence on the specialist's visual experience.

Keywords: computer vision systems, convolutional neural networks, dental diagnosis, Faster R-CNN, PyTorch, Anaconda, Jupyter Notebook, Data Augmentation, caries, periapical lesions.

ÍNDICE

I.	Introducción	1
II.	PROBLEMA DE ESTUDIO	2
III.	Justificación	3
IV.	Objetivos	5
IV-A.	Objetivo general	5
IV-B.	Objetivos específicos	5
V.	Fundamentos Teóricos	6
V-A.	Salud Bucal	6
V-B.	Redes neuronales convolucionales.	6
V-B1.	Estructura de conexiones.	7
V-C.	Conceptos Fundamentales de Machine Learning	7
V-C1.	Machine Learning	7
V-C2.	Modelos principales de aprendizaje automático	8
V-D.	Aprendizajes de filtros de convolución.	10
V-E.	Avances en Sistemas de Análisis de Imágenes de Rayos X dentales	10
VI.	Introducción al Análisis de Imágenes Dentales	11
VII.	Técnicas de Imagen Dental	11
VII-A.	Radiografías Panorámicas	11
VII-A1.	Ventajas	11
VII-A2.	Desventajas	12
VII-B.	Radiografías Periapicales	12
VII-B1.	Ventajas	12
VII-B2.	Desventajas	12
VIII.	Afecciones dentales principales.	12
VIII-A.	Caries Dental	12
VIII-B.	Lesiones Periapicales	13
VIII-C.	Diente Impactado	14
VIII-D.	Implantes Dentales	15
IX.	Inteligencia Artificial en el Análisis de Imágenes Dentales	16
IX-A.	Redes Neuronales Convolucionales (CNN)	16
IX-A1.	Arquitectura y Funcionamiento	16
IX-B.	Modelos de Detección de Objetos	16
IX-B1.	Faster R-CNN	16
IX-B2.	YOLO (You Only Look Once)	16
X.	Generación de Imágenes Sintéticas	17
X-A.	Generative Adversarial Networks (GANs)	17
X-A1.	Aplicaciones en el Aumento de Datos	17
X-B.	Técnicas Avanzadas en Generación de Imágenes	17
X-C.	DCGAN y StyleGAN	17
X-D.	Desafíos en el Análisis Automatizado de Imágenes Dentales	17
X-E.	Variabilidad en la Calidad de las Imágenes	17

X-F.	Preprocesamiento de Imágenes	17
X-G.	Anotación de Datos	17
X-H.	Herramientas de Anotación	17
XI.	Herramientas y Entornos de Desarrollo	18
XI-A.	Anaconda	18
XI-B.	Gestión de Paquetes y Entornos	19
XII.	Marco Metodológico	20
XII-A.	Diseño General	20
XII-B.	Tipo de Investigación	20
XII-C.	Nivel de Investigación	20
XII-D.	Diseño de Investigación	20
XII-E.	Método de Investigación	20
XII-F.	Población y Muestra	20
	XII-F1. Población	20
XII-G.	Muestra	20
XII-H.	Técnicas de Recolección de Datos	21
XII-I.	Herramientas y software:	21
XIII.	Procedimiento Metodológico	21
XIV.	Análisis de Datos	21
XIV-A.	Implementación del modelo.	21
XV.	Entrenamientos	24
XV-A.	Selección del entorno de trabajo y herramientas	24
XV-B.	Dataset utilizado	26
XV-C.	Modelo utilizado	27
XV-D.	Procedimiento de entrenamiento	27
XV-E.	Análisis de los resultados y errores iniciales.	28
XVI.	Entrenamiento del Modelo Faster R-CNN: Análisis Detallado	28
XVI-A.	Modelo Utilizado	28
XVI-B.	Prueba técnica 1	28
XVI-C.	Prueba técnica 2	30
XVI-D.	Prueba técnica 3	31
XVI-E.	Prueba técnica 4	33
XVI-F.	Prueba técnica 5	34
XVI-G.	Prueba técnica 6	37
XVI-H.	Prueba técnica 7	38
XVI-I.	Prueba técnica 8	40
XVI-J.	Prueba técnica 9	41
XVI-K.	Prueba técnica final	44
XVII.	Resultados	46
XVII-A.	Precisión y Aprendizaje del Modelo	46
XVII-B.	Impacto del Aumento de Datos (Data Augmentation)	46
XVII-C.	Oportunidades de Mejora	46
XVII-D.	Perspectiva Futura	46
XVII-E.	Conclusión Técnica	46

XVIII.Cronograma	47
XIX. PRESUPUESTO	48
XX. Conclusiones	49
XXI. Recomendaciones	50
Referencias	51
Anexo A: Proyecto	53

ÍNDICE DE FIGURAS

1.	Aplicación dental de IA	3
2.	Red neuronal monocapa - Red neuronal multicapa.	6
3.	Redes neuronales por realimentación	7
4.	Redes neuronales artificiales	7
5.	Aprendizaje automático.	8
6.	Aprendizaje supervisado.	9
7.	Aprendizaje no supervisado.	10
8.	Imágenes panorámicas clínica marisori - dataset	11
9.	Imágenes panorámicas periapicales - dataset	12
10.	Caries dental	13
11.	Lesion periapical.	14
12.	Diente impactado de dataset.	14
13.	implante de dataset.	15
14.	Inteligencia artificial en odontología	16
15.	Labelme - programa de etiquetas	18
16.	Programa principal - Anaconda	18
17.	Entorno virtual - Anaconda	19
18.	Entorno virtual creado - librerías	19
19.	Entorno para ejecutar código - Jupyter	21
20.	Entorno de entrenamiento por C. Franco, google colab.	25
21.	Entorno de entrenamiento 2 por C. Franco, google colab.	26
22.	Entrenamientos previos, por C. Franco Jupyter notebook.	27
23.	Resultado mAP de la prueba 1 con 4 clases más una de fondo.	29
24.	Pérdida durante el entrenamiento en la prueba 1.	29
25.	Curva de pérdida durante el entrenamiento de la Prueba 2	30
26.	Curva de mAP durante el entrenamiento de la Prueba 2	31
27.	Curva de pérdida durante el entrenamiento de la Prueba 3	32
28.	Curva de mAP durante el entrenamiento de la Prueba 3	32
29.	Curva de pérdida durante el entrenamiento de la Prueba 4	33
30.	Curva de mAP durante el entrenamiento de la Prueba 4	34
31.	Curva de pérdida durante el entrenamiento de la Prueba 5	35
32.	Curva de mAP durante el entrenamiento de la Prueba 5	36
33.	Curva de pérdida durante el entrenamiento de la Prueba 6	37
34.	Curva de mAP durante el entrenamiento de la Prueba 6	38
35.	Curva de pérdida durante el entrenamiento de la Prueba 7	39
36.	Curva de mAP durante el entrenamiento de la Prueba 7	39
37.	Resultados para caries, Jupyter notebook.	40
38.	Curva de pérdida durante el entrenamiento de la Prueba 8	40
39.	Curva de mAP durante el entrenamiento de la Prueba 8	41
40.	Imágenes panorámicas aumentadas en prueba 9	42
41.	Curva de pérdida durante el entrenamiento de la Prueba Final	42
42.	Curva de mAP durante el entrenamiento de la Prueba 9	43
43.	Curva de pérdida durante el entrenamiento de la Prueba final	44
44.	Curva de mAP durante el entrenamiento de la Prueba final	45
45.	Resultados de entrenamiento de la Prueba final.	45
46.	Verificación de GPU habilitada por C. Franco, Jupyter Notebook.	53
47.	Instalación de librerías necesarias por C. Franco, Jupyter Notebook.	53
48.	Comprobación de GPU habilitada por C. Franco, Jupyter Notebook.	54
49.	Instalación de herramientas por C. Franco, Jupyter Notebook.	54

50.	Instalación de métricas por C. Franco, Jupyter Notebook.	55
51.	Importación de librerías para entrenamiento por C. Franco, Jupyter Notebook.	56
52.	código de métricas 1 por C. Franco, Jupyter Notebook.	56
53.	código de métricas 2 por C. Franco, Jupyter Notebook.	57
54.	Código dataset 1 por C. Franco, Jupyter Notebook.	58
55.	Código dataset 2 por C. Franco, Jupyter Notebook.	59
56.	Clasificación de etiquetas 1 por C. Franco, Labelme.	60
57.	Etiquetado incorrecto demostración de etiqueta rectangular por C. Franco, Labelme.	61
58.	Etiquetado correcto demostración de etiqueta rectangular por C. Franco, Labelme	62
59.	Imágenes panorámicas con etiquetas por C. Franco, dataset	63
60.	Imágenes periapical por C. Franco, dataset	63
61.	Imágenes periapical 2 con etiquetas por C. Franco, dataset.	64
62.	Etiquetado poligonal para todo tipo de dientes, por C. Franco, Labelme	64
63.	Etiquetado poligonal 2 para todo tipo de dientes, por C. Franco, Labelme	65
64.	Etiquetado rectangular imagen panorámica, por C. Franco, Labelme.	65
65.	Etiquetado rectangular imagen periapical, por C. Franco, Labelme.	66
66.	Etiquetado rectangular imagen panoramica, por C. Franco, Labelme.	66
67.	Pruebas realizadas, por C. Franco, conda.	67
68.	Etiquetado de imágenes con el profesional	68

ÍNDICE DE TABLAS

I.	Afecciones dentales en Ecuador	2
II.	Cronograma	47

I. INTRODUCCIÓN

El análisis de imágenes de rayos X dentales representa un componente crucial en la práctica odontológica, facilitando la detección de afecciones que no son visibles a simple vista. Sin embargo, la interpretación manual de estas imágenes puede ser subjetiva y propensa a errores, especialmente dado el volumen y la complejidad de los datos radiográficos. En este contexto, el desarrollo de sistemas automatizados para el análisis de imágenes se vuelve indispensable para mejorar la precisión y eficiencia del diagnóstico dental.

Este estudio se enfoca en el diseño y la implementación de un sistema automatizado basado en redes neuronales convolucionales (CNN) para la detección de afecciones dentales. Utilizando un conjunto de datos de radiografías dentales anotadas, el proyecto se propone desarrollar un modelo que pueda identificar y clasificar estas afecciones con alta precisión. La elección del modelo Faster R-CNN, conocido por su eficacia en tareas de detección de objetos, responde a la necesidad de un análisis detallado y preciso de las imágenes radiográficas.

Para la implementación de este sistema, se ha optado por PyTorch como framework de desarrollo debido a su flexibilidad y robustez. Jupyter Notebook se ha utilizado para la escritura y ejecución del código, facilitando la experimentación y el ajuste del modelo, mientras que Anaconda ha sido clave en la gestión del entorno de desarrollo y las dependencias. Además, se han integrado técnicas de Data Augmentation para ampliar y diversificar el conjunto de datos, lo que contribuye a una mayor generalización del modelo ante nuevas imágenes.

El objetivo final de este sistema es su aplicación en entornos clínicos, donde puede servir como una herramienta de apoyo para los odontólogos, mejorando la precisión del diagnóstico y optimizando el tiempo requerido para la evaluación de las imágenes.

II. PROBLEMA DE ESTUDIO

Las enfermedades orales representan una problemática global, especialmente en países con recursos limitados, afectando tanto la salud general como generando un impacto económico significativo.[1].A pesar de los esfuerzos dirigidos a abordar patologías bucales como caries dental y enfermedad periodontal, persisten desafíos, que pueden resultar en tiempo limitado de diagnósticos, provocando incertidumbre y errores en la búsqueda de tratamientos especializados por parte de los pacientes. La falta de claridad y orientación resalta la necesidad de soluciones innovadoras como la automatización de procesos.[2]

Estudios epidemiológicos realizados en los últimos 20 años demuestran una alta prevalencia de problemas bucales, incluyendo caries dental, enfermedades periodontales y maloclusión.A pesar de los esfuerzos implementados por el sistema de salud, la necesidad de tratamiento odontológico y la prevalencia de procedimientos como restauraciones dentales y endodoncias destacan la magnitud del desafío en la atención bucal.[1].

Otro de los desafíos reside en la falta de pericia que puede enfrentar un profesional recién graduado en odontología al diagnosticar afecciones bucales. Los dentistas, al depender en gran medida de la observación visual, desarrollan esta habilidad con la experiencia. Esta carencia de experiencia podría dificultar la identificación rápida y precisa de la afectación bucal, lo que podría resultar en demoras al momento de diagnosticar y tratamientos incorrectos que agraven la condición de los pacientes. La demora en los diagnósticos odontológicos puede tener un impacto significativo en la salud bucal de los pacientes[3].

A pesar del progreso experimentado en el ámbito de la salud bucal en Ecuador, persisten desafíos notables, especialmente en lo que respecta a la implementación y actualización en los niveles de atención. Se identifican áreas donde la atención sigue siendo individualista, compleja, costosa y vinculada a la tecnología, la cual sigue siendo deficiente, presentando además, una cobertura limitada y sin abordar el enfoque integral.La falta de herramientas eficientes para el análisis de imágenes de rayos X puede contribuir a esta demora, lo que a su vez puede resultar en un tratamiento inadecuado. La necesidad persistente de intervenciones efectivas y la falta de herramientas automatizadas para la detección temprana motivan la propuesta de este proyecto.[4]

Cuadro Estadístico sobre la Salud Bucal en Ecuador

Tabla I
AFECCIONES DENTALES EN ECUADOR

Enfermedad oral	Prevalencia
Caries Dentales	75,6 %
Enfermedad Periodontal	20-50 % (generales) 70 % (adultos mayores de 65 años, 56,4 % en hombres)
Maloclusión Dental	35 % (leve), 13.5 % (moderada/severa, escolares de 12 a 15 años) 29-70 % (Clase I), 10-34 % (Clase II), 10-19 % (Clase III, 20-27 años)
Fluorosis	5 % (escolares de 12 a 15 años)
Procedimientos de Operacion	85,2 %
Exodoncias	16,9 %
Endodoncia	10,5 %

III. JUSTIFICACIÓN

En el ámbito odontológico, la inteligencia artificial (IA) desempeña un papel fundamental para perfeccionar la exactitud en el proceso de diagnóstico y optimización de tiempo. Este sector, que es relativamente reciente, experimentó un progreso significativo en los últimos años. Esto incluye el empleo de IA para analizar imágenes radiográficas y tomográficas. Estos sistemas tienen la capacidad de detectar anomalías y patrones que podrían escapar a la atención de los dentistas; detectan problemas como caries, fracturas o enfermedad periodontal, mejorando la precisión del diagnóstico y permitiendo detectar afecciones antes de que se agraven, lo que da como resultado una mejor atención del paciente.[5].

Cuadro de aplicación de IA en diagnósticos odontológicos:

Tabla 4. Aplicación dental del AI en diagnóstico, cariológia y endodoncia

Autor	País y año	Unidad de estudio	Modalidad	Imágenes totales	Tipo de inteligencia artificial	Resultado principal	Precisión (p), sensibilidad (s), especificidad (e)
Hiraiva <i>et al.</i> ⁵	Japón 2019	Morfología radicular de las primeras molares mandibulares	CBCTRx.panorámicas	760	CNN	Alta precisión en el diagnóstico diferencial de las raíces distales de los primeros molares mandibulares	86,9% (P)
Lee <i>et al.</i> ⁶⁰	Corea 2018	Caries	Rx. periapicales	3000	CNN architecture (GoogLeNet Inception v3)	Alto potencial en la detección y el diagnóstico de la caries dental	89,0% (P)
Mallishery <i>et al.</i> ³²	Alemania 2019	Dificultad del tratamiento endodontico	Pacientes de endodoncia	500	DNN	Evaluación rápida del nivel de dificultad de los tratamientos endodónticos y correcta derivación al especialista	94,96% (S)
Yamaguchi <i>et al.</i> ⁶¹	Japón 2019	Desprendimiento de las coronas de resinas compuestas CAD/CAM	Imágenes 3D	8640	CNN	Rendimiento bueno de la probabilidad de despegado de una corona de RC CAD/CAM	97,0% (P)
Tuzoff <i>et al.</i> ⁶²	Rusia 2019	Identificación de dientes	Rx. panorámica	1352	CNN architecture (VGG-16)	Simplificación de la detección y enumeración de dientes	0,9941 (S)0,9945 (P)
Javed <i>et al.</i> ⁶³	Arabia 2019	Presencia del post-Streptococcus mutans	Molares primaries	45	ANN	Predicción eficiente de la presencia de post-Streptococcus mutans	-
Hung <i>et al.</i> ³⁰	EEUU. 2019	Caries radicular	Ficha de datos	9544	SL	Intervención temprana y tratamiento de la caries radicular	95,1% (P)99,6% (S)99,6% (E)
Casalegno <i>et al.</i> ⁶⁴	Suiza 2019	Caries	Imágenes IR	185	CNN	Mayor velocidad y precisión en la detección de caries	-
Yasa <i>et al.</i> ⁶⁵	Turquía 2020	Identificación de dientes	Rx. de aleta de mordida	1125	R-CNN	Rápida detección y enumeración de dientes	0,9293 (P)0,9748 (S)
Zhang <i>et al.</i> ⁶⁶	China 2020	Caries	Fotos orales	3932	CNN	Mejora la detección de caries en fotos orales	81,90% (S)
Schwendicke <i>et al.</i> ³¹	Alemania 2020	Caries proximal	Rx. de aleta de mordida	3293	CNN architecture (U-Net)	Mayor rentabilidad en la detección temprana de caries	0,80 (P)0,75 (S)0,83 (E)
Setzer <i>et al.</i> ⁵³	EEUU. 2020	Lesiones periapicales	CBCT	20	CNN architecture (U-Net)	Alta precisión en la detección de lesiones	0,93 (P)0,88 (E)
Abdalla-Aslan <i>et al.</i> ⁶⁷	Israel 2020	Restauraciones dentales	Rx. panorámica	83	ML	Excelente rendimiento en la detección y clasificación de restauraciones	93,6% (P)
Jeon <i>et al.</i> ⁵⁵	Corea 2021	Conductos en forma de C en segundos molares mandibulares	CBCT y Rx. panorámica	2040	CNN architecture (Xception)	Precisión en la predicción de conductos en forma de C	92,7 (S)97,0 (E)95,9% (P)
Ekert <i>et al.</i> ⁵⁴	Alemania 2019	Lesiones apicales (LA)	Rx. panorámica	85	CNN	Capacidad discriminatoria para detectar LA	0,65 (S)0,87 (E)
Leite <i>et al.</i> ⁶⁸	Alemania 2020	Identificación de dientes	Rx. panorámica	153	CNN	Mayor rapidez y precisión para la detección y segmentación de dientes	98,9% (S)99,6% (P)

Figura 1. Aplicación dental de IA

Fuente: Aplicación de la inteligencia artificial en Odontología - artículo. [6].

Se evidencia que la mayoría de los estudios se enfocan en siete áreas fundamentales dentro de distintas especialidades odontológicas. Estas áreas incluyen odontología forense, medicina, periodoncia, implantología oral, patología oral, ortodoncia, cirugía maxilofacial, diagnóstico, cariológia y endodoncia. Respecto a la evaluación de las técnicas empleadas, se destaca que las radiografías panorámicas son las más frecuentes, seguidas por las radiografías periapicales y la tomografía computarizada. Se identificó que la especialidad con mayor cantidad de trabajos relacionados con inteligencia artificial fue la de diagnóstico, cariológia y endodoncia. Por lo tanto, se plantea la necesidad de desarrollar un sistema de análisis de radiografías específico para profesionales y estudiantes en Ecuador, enfocado en cariológia y lesiones periapicales. Este sistema tiene como propósito mejorar la eficiencia en los diagnósticos y ofrecer una mayor precisión en la identificación de enfermedades dentales en los pacientes.[6]

La actual generación de profesionales odontólogos se forman y desempeñan su labor con dispositivos computarizados en un entorno virtual, adaptándose al uso de estos recursos de manera fluida y natural.[7] La IA y la

experiencia clínica combinadas prometen una mejora innovadora en odontología. Estos sistemas de análisis basados en IA, junto con la experiencia del especialista, potencian la precisión del diagnóstico y su rapidez. Esta sinergia entre tecnología y experiencia clínica mejora las decisiones de tratamiento y agiliza la atención precisa a los pacientes.

IV. OBJETIVOS

IV-A. Objetivo general

Desarrollar un sistema de análisis de radiografías dentales a través de visión artificial para la detección de afecciones dentales como la caries y las lesiones periapicales. Que mejore la precisión de diagnósticos en el área odontológica.

IV-B. Objetivos específicos

- Recolectar un conjunto de datos de radiografías dentales con diversas enfermedades, para el desarrollo de una red neuronal convolucional que permita el reconocimiento de las afecciones dentales.
- Entrenar la red neuronal convolucional con un conjunto de datos de prueba para la detección y clasificación de enfermedades dentales según el tipo de afectación.
- Evaluar el sistema utilizando métricas de desempeño, donde se evidencie su efectividad mediante la comparación entre los diagnósticos generados y las predicciones generadas por la red neuronal convolucional.

V. FUNDAMENTOS TEÓRICOS

V-A. Salud Bucal

El informe más reciente sobre el estado de la salud bucal en todo el mundo publicado por la Organización Mundial de la Salud (OMS) proporciona una imagen completa de la cantidad de enfermedades dentales con datos específicos de 194 países.

Según los datos presentados , casi la mitad de la población mundial (4,5 millones de personas, o 3,5 millones de individuos) registran problemas de salud bucal , y tres de cada cuatro personas afectadas viven en países de ingresos bajos o medios a nivel global, lo cual indica la falta de acceso a la prevención y tratamiento adecuado en este campo de salud. El informe describe varias oportunidades prometedoras para mejorar el estado general de la salud bucal , entre ellas:

- Adoptar una perspectiva de salud pública al abordar elementos de riesgo comunes y fomentar una alimentación equitativa.
- Planificación de servicios en las operaciones nacionales de atención de salud .
- Expandir las habilidades de los trabajadores de salud no especializados en odontología para expandir la cobertura de los servicios en este campo y redefinir los modelos de atención del personal en salud oral para cumplir con las necesidades de la población, esto implica realizar mejoras en el sistema automatizado para un diagnóstico más rápido y preciso .
- Sistemas de información mediante la recopilación e integración de datos de salud bucal en los sistemas nacionales de seguimiento de la salud[8].

V-B. Redes neuronales convolucionales.

Las Redes Neuronales Convolucionales (CNN) se originaron a partir del neocognitrón desarrollado por Fukushima. Su concepto se fundamenta en el estudio biológico del reconocimiento visual primario en la corteza humana investigado por Hubel y Wiesel.[9]

Se clasifican según el número de capas, grado de conexión y realimentación.[10].

Redes neuronales monocapa y multicapa.



Figura 2. Red neuronal monocapa - Red neuronal multicapa.
Fuente:<https://inteligenciafutura.mx/m>

Redes neuronales según realimentación.

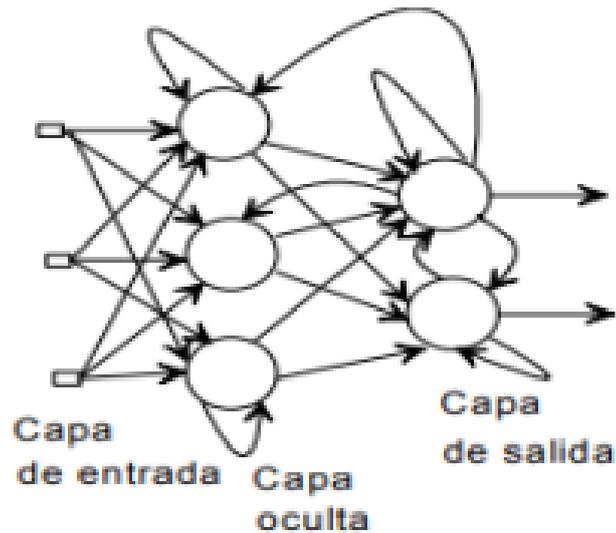


Figura 3. Redes neuronales por realimentación

Fuente: <https://idus.us.es>

V-B1. *Estructura de conexiones.*: Las neuronas artificiales, surgen a partir de la adición de las entradas multiplicadas por sus respectivos pesos, lo cual juega un rol crucial en la determinación del impulso nervioso que llega a la neurona. Este valor, tras ser procesado dentro de la célula a través de una función de activación, produce un resultado que se utiliza como salida de la neurona.[11]

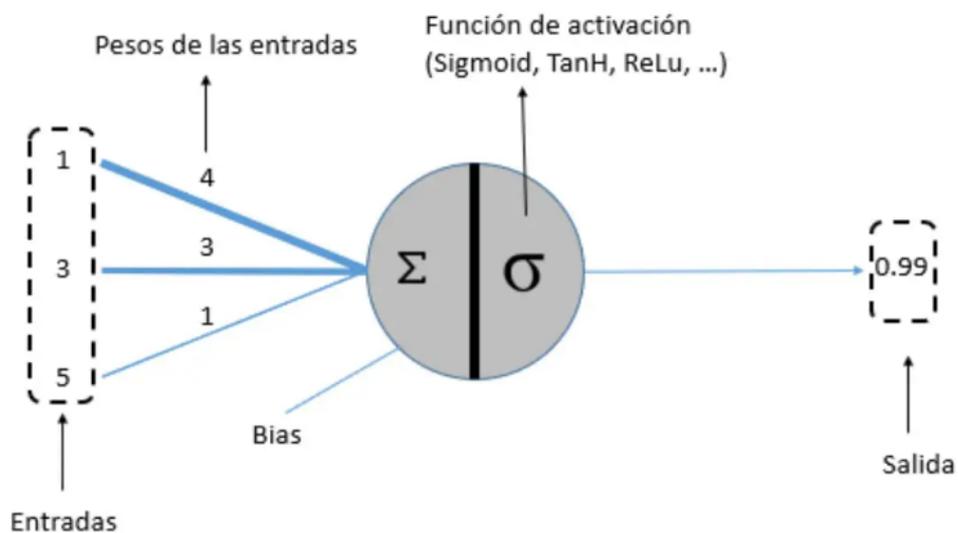


Figura 4. Redes neuronales artificiales

Fuente: <https://www.xeridia.com/blog/redes-neuronales-artificiales-que-son-y-como-se-entrenan-parte-i>

V-C. Conceptos Fundamentales de Machine Learning

V-C1. *Machine Learning*: El término “machine learning” o aprendizaje de automático fue acuñado en 1959 por Arthur Samuel, una figura pionera en el campo. Su contribución fue considerable al desarrollar un

programa informático para el juego de damas, el cual tenía la capacidad de evaluar diversas opciones de movimiento y almacenar las posiciones previas. Esta capacidad permitiría que la máquina aprendiera y mejorara su desempeño con cada juego. En su publicación sobre IA, él describe el aprendizaje automático como una tecnología que hace uso de técnicas estadísticas y algoritmos computacionales. Debido a este enfoque las computadoras son capaces de aprender y mejorar su productividad en tareas puntuales, conforme procesan una cantidad de datos, prescindiendo de instrucciones explícitas facilitadas por el programador.

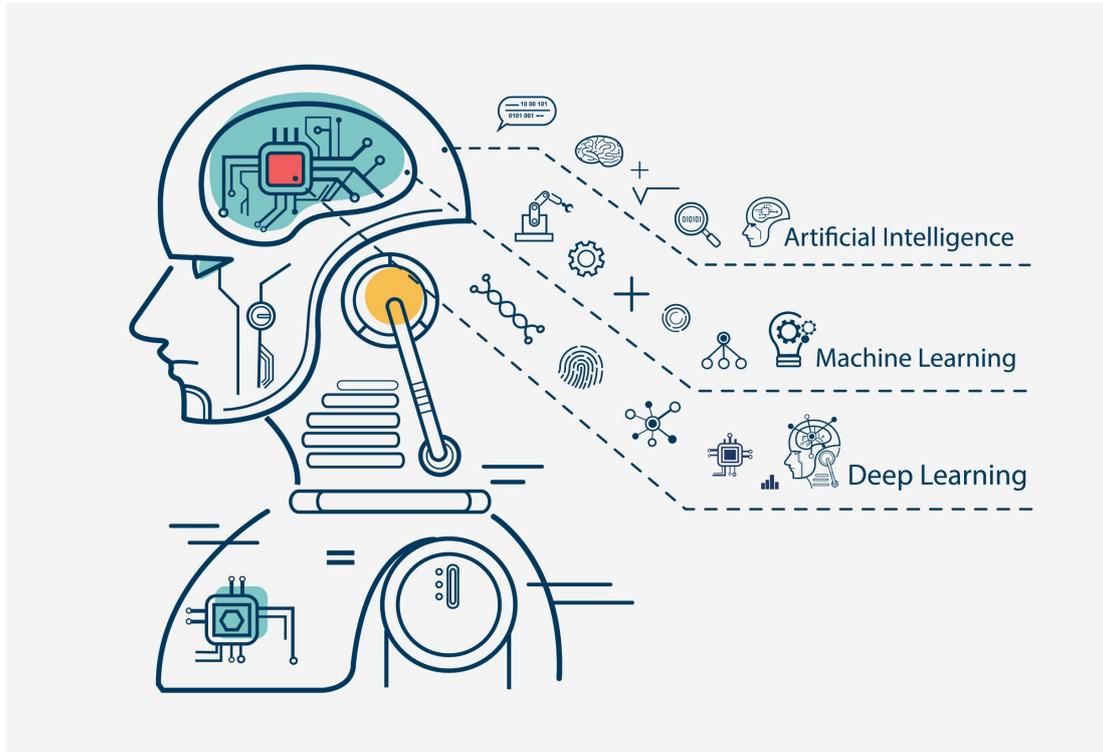


Figura 5. Aprendizaje automático.
Fuente: <https://inteligenciafutura.mx/ml>

El aprendizaje automático se considera una subdivisión de la inteligencia artificial debido a su capacidad para aprender mediante el análisis de datos y tomar decisiones basadas en algoritmos. La conexión entre ambas disciplinas se destaca cómo una herramienta fundamental dentro del campo más amplio de la inteligencia artificial, ya que aprovecha la capacidad de las máquinas para mejorar su rendimiento a partir de la experiencia de aprendizaje automático.[12]

Russell y Norving postularon que la finalidad de la Inteligencia Artificial (IA) debería consistir en la creación de máquinas capaces de emular comportamientos inteligentes, las cuales podrían ser categorizadas en distintos grados. Estas clasificaciones abarcarían sistemas capaces de imitar el pensamiento y la acción humana, así como sistemas capaces de actuar y razonar de manera lógica. Asimismo, se anticipa que estas máquinas puedan llevar a cabo una variedad de funciones, como el procesamiento de lenguaje natural, la representación del conocimiento, el razonamiento automático, el aprendizaje automático, la visión computacional y la robótica, todas las cuales han tenido un papel fundamental en la evolución.[13].

V-C2. Modelos principales de aprendizaje automático: Existen dos enfoques fundamentales en el aprendizaje automático: el supervisado y el no supervisado.

El aprendizaje supervisado implica entrenar un algoritmo de Machine Learning proporcionándole tanto las preguntas (características) como las respuestas asociadas (etiquetas). Esta metodología permite que, en el futuro, el algoritmo realice predicciones basadas en las características conocidas durante el entrenamiento.

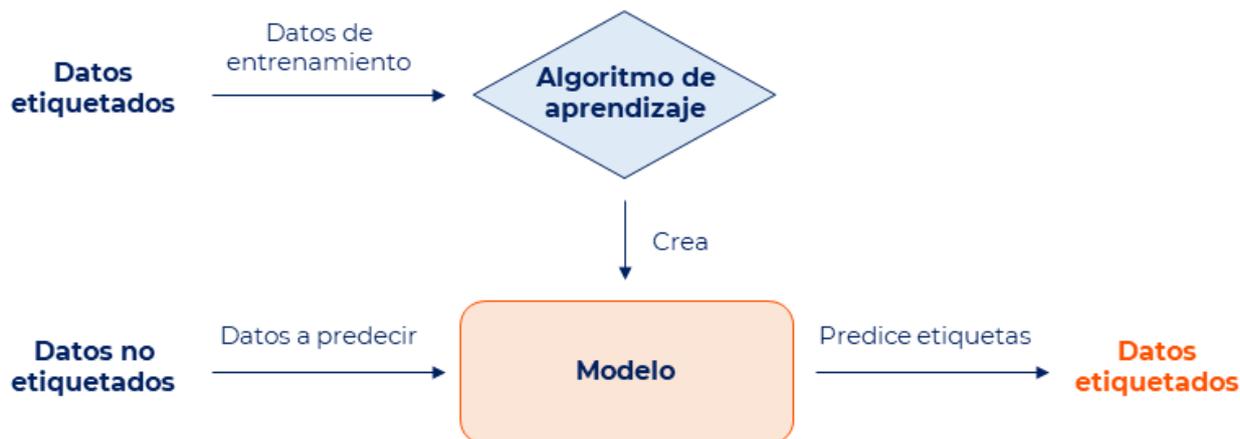


Figura 6. Aprendizaje supervisado.
Fuente: <https://www.diegocalvo.es/>.

Dentro del aprendizaje supervisado, se distinguen dos tipos principales de algoritmos de entrenamiento: clasificación y regresión. Estos enfoques se utilizan respectivamente para problemas en los que se busca clasificar datos en categorías discretas o para aquellos en los que se desea predecir valores continuos.

El algoritmo de clasificación tiene como finalidad determinar a qué grupo o categoría corresponde un elemento característico bajo análisis. Este algoritmo identifica patrones dentro de los datos suministrados y los utiliza para clasificar los elementos en grupos específicos.

Posteriormente, cuando se le presentan nuevos datos, el algoritmo los compara con estos patrones previamente identificados para asignarlos a uno de los grupos, permitiendo así realizar predicciones sobre su naturaleza.

La variable que se busca predecir en este tipo de algoritmo consiste en un conjunto de estados discretos o categóricos. Estos estados pueden presentarse en distintas formas:

- Variables binarias, como por ejemplo Sí, No, Azul, Rojo, Fuga, No Fuga, entre otras.
- Variables múltiples, representadas por opciones como Compraré producto1, producto 2, etc, y similares.
- Variables ordenadas, tal como Riesgo Bajo, Medio, Alto, entre otras posibilidades.[14]

Es posible dividir el aprendizaje supervisado en:

- Aprendizaje por corrección de error: es decir la diferencia entre los valores esperados y los obtenidos.
- Aprendizaje estocástico: realiza cambios aleatorios quedándose finalmente con los mejores resultados.

En el aprendizaje no supervisado no se proporciona ningún dato extra al sistema además de los datos de entrada, y la tarea que debe desarrollar es la de extraer datos con características relevantes. El método que se aplicará en este trabajo es el aprendizaje supervisado, ya que disponemos de datos etiquetados que queremos que el modelo aprenda.

- Aprendizaje hebbiano: Mide el nivel de familiaridad y extrae los datos de entrada.
- Aprendizaje competitivo y comparativo: Implica agregar elementos a una clase, ajustando los pesos si se determina que el nuevo elemento pertenece a dicha clase; de lo contrario, se tiene la opción de crear una nueva clase.
- Aprendizaje por refuerzo: Menos ágil que el aprendizaje basado en corrección de errores, en esta modalidad no se cuenta con un conjunto completo de datos de salida precisos, sino que únicamente se indica si el dato es considerado válido o no.[15]

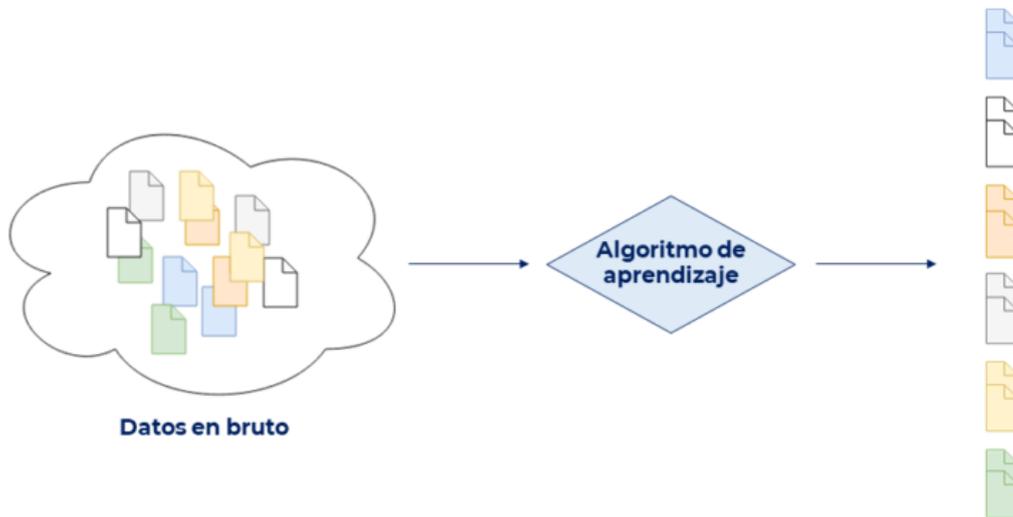


Figura 7. Aprendizaje no supervisado.
Fuente: <https://www.diegocalvo.es/>.

V-D. Aprendizajes de filtros de convolución.

De acuerdo con la fórmula, se establece que la convolución de un volumen con una profundidad D necesita D filtros para generar un volumen con $D = 1$. Esto implica que la convolución extrae una única característica del volumen inicial, que, dependiendo del caso, podría no ser suficiente como vector de características único para un algoritmo de Machine Learning. Las características obtenidas tras la operación de convolución se organizan espacialmente, representando mapas de características en lugar de vectores de características directos. Para convertir un mapa $O \times O$ en un vector O^2 , se puede linealizar.

Para necesidades básicas, como detectar líneas horizontales, verticales o un color específico, una sola característica puede ser adecuada. No obstante, en la investigación de patrones más complejos, como identificar una persona o una máquina, una sola característica resulta insuficiente para abarcar la complejidad y variabilidad (posiciones, dimensiones o colores) de la imagen. La definición manual de filtros para extraer estas características es un proceso arduo y complicado, con resultados variables. Por esta razón, el aprendizaje automático (Machine Learning) se emplea para aprender los valores de los filtros, permitiendo que la combinación de las características extraídas de estos filtros logre caracterizar completamente la imagen en estudio.

V-E. Avances en Sistemas de Análisis de Imágenes de Rayos X dentales

El uso de la IA en radiología está creciendo, particularmente en el desarrollo laboral, donde puede mejorar la eficiencia de las solicitudes y ofrecer resultados más consistentes. La tecnología Eclipse de Carestream utiliza IA para mejorar el flujo de trabajo de los solicitudes, al mismo tiempo que mejora la calidad y la consistencia en las

imágenes diagnósticas.[16] Esta es una tecnología que con el tiempo dará un valor agregado a los diagnósticos médicos, cambiando el paradigma de lo que hasta ahora se conoce. La aplicación de IA en odontología es un hecho, tal es el ejemplo del sistema de Pearl, que ayuda a los profesionales en su revisión de radiografías. Esto proporciona a los odontólogos una perspectiva más clara del estado de la salud bucal del paciente.[17]

VI. INTRODUCCIÓN AL ANÁLISIS DE IMÁGENES DENTALES

El análisis de imágenes dentales es esencial para el diagnóstico y tratamiento de diversas afecciones dentales. Las imágenes radiográficas, como las panorámicas y periapicales, proporcionan una vista detallada de la dentadura y los tejidos circundantes. La evolución tecnológica en este campo ha llevado a la incorporación de técnicas avanzadas de procesamiento de imágenes y aprendizaje automático, mejorando significativamente la precisión y la eficiencia del diagnóstico [18].

VII. TÉCNICAS DE IMAGEN DENTAL

VII-A. Radiografías Panorámicas

Las radiografías panorámicas ofrecen una vista amplia de la dentadura, incluyendo dientes, huesos y tejidos blandos. Esta técnica es útil para una evaluación general, el diagnóstico de afecciones óseas y la planificación de tratamientos complejos como la colocación de implantes. Sin embargo, pueden presentar limitaciones en la resolución y detalle específico de áreas pequeñas, lo que a veces puede complicar el diagnóstico de afecciones dentales más sutiles [19].



Figura 8. Imágenes panorámicas clínica marisori - dataset

VII-A1. Ventajas:

- Cobertura completa de la dentadura en una sola imagen.
- Útil para la planificación de tratamientos quirúrgicos.
- Permite una evaluación rápida del estado general de la salud dental.

VII-A2. Desventajas:

- Menor resolución en áreas específicas.
- Riesgo de superposiciones y artefactos que pueden complicar el diagnóstico.
- Menos efectiva para la detección temprana de caries o lesiones pequeñas.

VII-B. Radiografías Periapicales

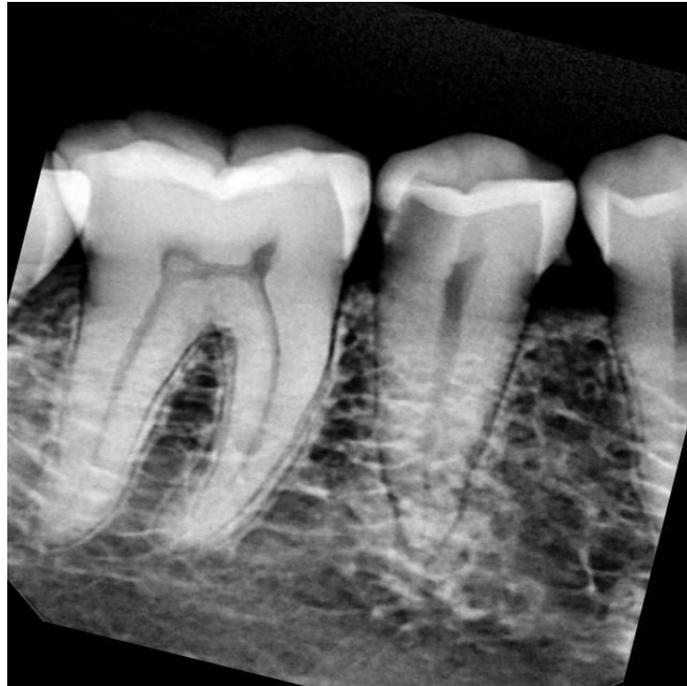


Figura 9. Imágenes panorámicas periapicales - dataset

Las radiografías periapicales se centran en una región específica de la dentadura, proporcionando una visión detallada de los dientes y los tejidos circundantes. Son particularmente efectivas para identificar caries, infecciones en el ápice dental y problemas óseos localizados. Esta técnica requiere la toma de múltiples radiografías para obtener una visión completa de toda la dentadura, lo que puede ser menos conveniente para el paciente y el profesional [19].

VII-B1. Ventajas:

- Alta resolución en áreas específicas.
- Ideal para detectar caries, infecciones y problemas óseos locales.
- Permite una evaluación detallada de dientes individuales.

VII-B2. Desventajas:

- Requiere la toma de varias imágenes para cubrir toda la dentadura.
- Menos útil para la evaluación general de la salud dental.

VIII. AFECIONES DENTALES PRINCIPALES.

VIII-A. Caries Dental

Las caries dentales son una de las afecciones bucales más comunes a nivel mundial, causadas principalmente por la interacción de los ácidos producidos por la descomposición de azúcares por bacterias presentes en la placa

dental. El avance de las caries puede llevar a la destrucción del esmalte dental y, eventualmente, afectar las capas más profundas del diente.



Figura 10. Caries dental

Fuente: <https://patoral.umayor.cl/radiologia-de-las-caries/>

La inteligencia artificial (IA) ha revolucionado la detección de caries, permitiendo una identificación más precisa y temprana en comparación con los métodos tradicionales de examen visual y radiográfico [20]. Los estudios recientes han resaltado el uso de selladores dentales y flúor como estrategias preventivas para reducir la prevalencia de caries, especialmente en niños y poblaciones vulnerables.

VIII-B. Lesiones Periapicales

Las lesiones periapicales son inflamaciones o infecciones que se desarrollan alrededor del ápice (punta) de la raíz de un diente debido a la necrosis pulpar. Estas lesiones pueden surgir como resultado de caries no tratadas, traumatismos o intervenciones dentales previas mal realizadas.

Los estudios recientes se han centrado en la efectividad del tratamiento endodóntico para controlar las lesiones periapicales, enfatizando el papel de la limpieza y desinfección del canal radicular en la prevención de la progresión de estas lesiones. Además, se ha reportado que el uso de redes neuronales convolucionales mejora significativamente la capacidad de identificar lesiones periapicales a través de radiografías dentales [21].



Figura 11. Lesion periapical.

Fuente: <https://dentalborras.com/periodontitis-apical-cronica-tratada-mediante-cirugia-periapical/>

VIII-C. Diente Impactado

Un diente impactado es aquel que no ha erupcionado en su posición normal dentro de la arcada dental, quedando atrapado total o parcialmente en el hueso o el tejido blando. Los dientes más comúnmente impactados son los terceros molares (muelas del juicio).

La literatura reciente destaca la importancia de la intervención quirúrgica temprana para evitar complicaciones asociadas con los dientes impactados, como infecciones, formación de quistes o daño a los dientes adyacentes. También se están utilizando técnicas avanzadas de imagen, como la tomografía computarizada de haz cónico (CBCT), para permitir un diagnóstico más preciso y la planificación quirúrgica detallada para la extracción de dientes impactados [22].



Figura 12. Diente impactado de dataset.

VIII-D. *Implantes Dentales*

Los implantes dentales son dispositivos médicos utilizados para reemplazar dientes perdidos. Se insertan quirúrgicamente en el hueso maxilar o mandibular para proporcionar soporte a prótesis dentales como coronas, puentes o dentaduras completas.

Investigaciones recientes han mostrado que la tasa de éxito de los implantes dentales sigue siendo alta, con una supervivencia del 95 % o más después de 10 años de colocación, siempre y cuando se mantenga una adecuada higiene oral y cuidado postoperatorio. Además, se están utilizando tecnologías como la impresión 3D y la planificación quirúrgica asistida por computadora para mejorar la precisión en la colocación de los implantes y reducir las complicaciones [23].



Figura 13. implante de dataset.



Figura 14. Inteligencia artificial en odontología

IX-A. Redes Neuronales Convolucionales (CNN)

Las Redes Neuronales Convolucionales (CNN) han transformado el análisis de imágenes mediante la capacidad de aprender características jerárquicas y realizar tareas de clasificación y detección con alta precisión. En odontología, las CNN se utilizan para identificar caries, lesiones periapicales y otras afecciones dentales a partir de imágenes radiográficas. La arquitectura de las CNN permite la extracción automática de características relevantes, lo que mejora la precisión del diagnóstico y reduce la dependencia de la intervención humana [24].

IX-A1. Arquitectura y Funcionamiento: Las CNN están compuestas por varias capas, incluyendo capas convolucionales, de agrupamiento y totalmente conectadas. Las capas convolucionales extraen características locales mediante la aplicación de filtros, mientras que las capas de agrupamiento reducen la dimensionalidad de las características extraídas. Las capas totalmente conectadas realizan la clasificación final basándose en las características extraídas [24].

IX-B. Modelos de Detección de Objetos

IX-B1. Faster R-CNN: Faster R-CNN es un modelo de detección de objetos que utiliza una Red de Propuestas de Regiones (RPN) para generar regiones de interés en las imágenes. La RPN se encarga de proponer posibles ubicaciones de objetos, que luego son refinadas y clasificadas por una red de detección. Este enfoque mejora la precisión y la velocidad de la detección de objetos en comparación con métodos anteriores [25].

IX-B2. YOLO (You Only Look Once): YOLO es un modelo de detección en tiempo real que divide la imagen en una cuadrícula y predice las bounding boxes y las clases de los objetos en cada celda. A diferencia de Faster R-CNN, que realiza la detección en dos etapas, YOLO realiza la detección en una sola etapa, lo que permite una

mayor velocidad y es adecuado para aplicaciones clínicas en tiempo real [26].

X. GENERACIÓN DE IMÁGENES SINTÉTICAS

X-A. *Generative Adversarial Networks (GANs)*

Los Generative Adversarial Networks (GANs) son una técnica de generación de imágenes sintéticas que utiliza dos redes neuronales: una red generadora y una red discriminadora. La red generadora crea imágenes sintéticas, mientras que la red discriminadora evalúa su realismo. Ambas redes se entrenan en un juego adversarial, mejorando progresivamente la calidad de las imágenes generadas [27].

X-A1. Aplicaciones en el Aumento de Datos: Los GANs pueden generar imágenes sintéticas para aumentar conjuntos de datos en el entrenamiento de modelos de IA. Esto es particularmente útil en el contexto dental, donde puede haber una cantidad limitada de imágenes de alta calidad. La generación de imágenes sintéticas permite mejorar la diversidad del conjunto de datos y entrenar modelos más robustos [28].

X-B. *Técnicas Avanzadas en Generación de Imágenes*

X-C. *DCGAN y StyleGAN*

DCGAN (Deep Convolutional GAN): Utiliza redes convolucionales profundas en la arquitectura GAN, lo que mejora la calidad de las imágenes generadas y permite una mayor estabilidad en el entrenamiento [29].

StyleGAN: Introduce el concepto de *estilo* en la generación de imágenes, permitiendo un control más preciso sobre las características generadas y mejorando la calidad visual de las imágenes sintéticas [30]

X-D. *Desafíos en el Análisis Automatizado de Imágenes Dentales*

X-E. *Variabilidad en la Calidad de las Imágenes*

La calidad de las imágenes radiográficas puede variar debido a factores como el equipo utilizado, la técnica de toma y el procesamiento de imágenes. Esta variabilidad puede afectar la precisión de los modelos de IA y requiere técnicas de preprocesamiento y normalización para mitigar el impacto de estas diferencias [31].

X-F. *Preprocesamiento de Imágenes*

El preprocesamiento incluye técnicas como la normalización del contraste, la reducción de ruido y la corrección de artefactos. Estas técnicas mejoran la calidad de las imágenes y facilitan el entrenamiento de modelos de IA más precisos [32].

X-G. *Anotación de Datos*

La anotación precisa de imágenes es crucial para el entrenamiento de modelos de IA. La anotación manual es laboriosa y propensa a errores, lo que puede afectar la calidad del entrenamiento. Herramientas automatizadas y semi-automatizadas están en desarrollo para mejorar la eficiencia y precisión de la anotación [33].

X-H. *Herramientas de Anotación*

Labelme: Una herramienta de anotación de imágenes que permite etiquetar objetos en imágenes mediante un formato JSON. Es útil para la preparación de datos en proyectos de visión por computadora [34].

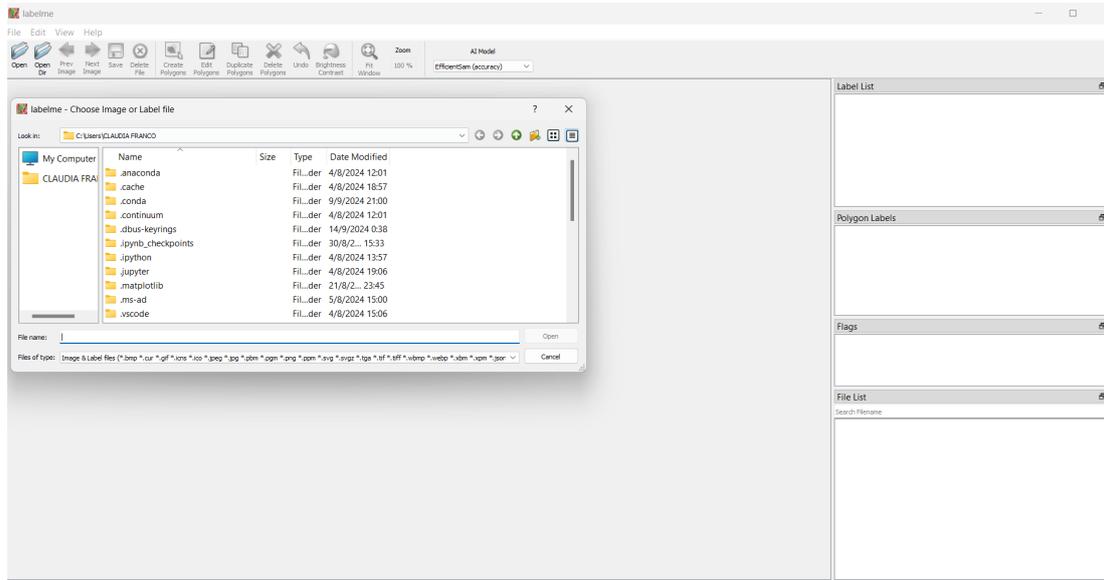


Figura 15. Labelme - programa de etiquetas

XI. HERRAMIENTAS Y ENTORNOS DE DESARROLLO

XI-A. Anaconda

Anaconda es una distribución de Python y R utilizada para la ciencia de datos y el aprendizaje automático. Proporciona un entorno integrado que facilita la gestión de paquetes y entornos de desarrollo. Incluye herramientas como Conda, Jupyter Notebook y Spyder, que son esenciales para la experimentación y el desarrollo de modelos de IA [35].

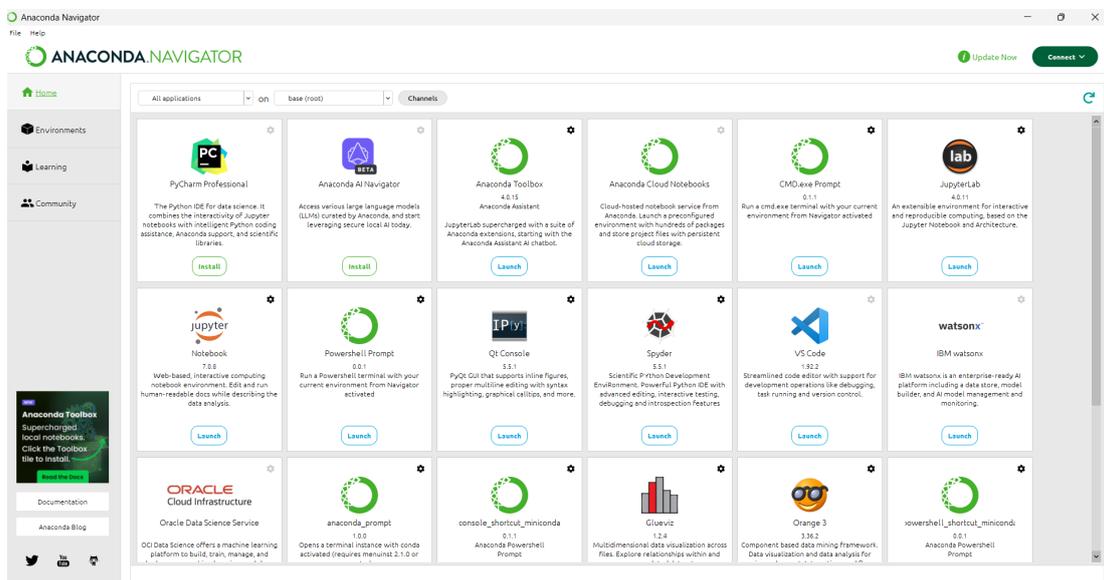


Figura 16. Programa principal - Anaconda

XI-B. Gestión de Paquetes y Entornos

Conda: Es un gestor de paquetes que permite la instalación y actualización de bibliotecas y herramientas necesarias para el desarrollo de modelos de IA. Facilita la creación de entornos virtuales para gestionar dependencias específicas de proyectos [36]

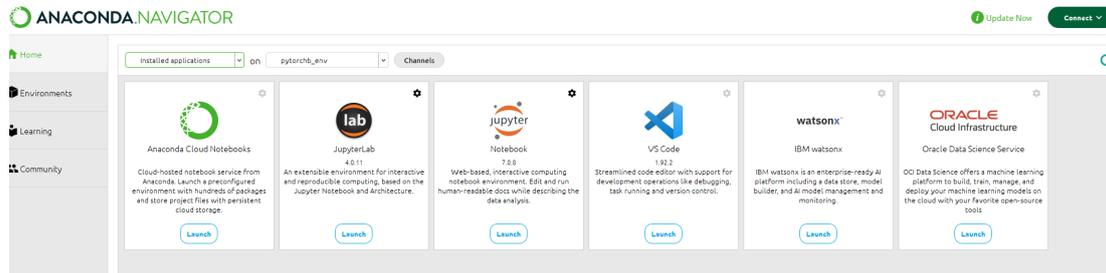


Figura 17. Entorno virtual - Anaconda

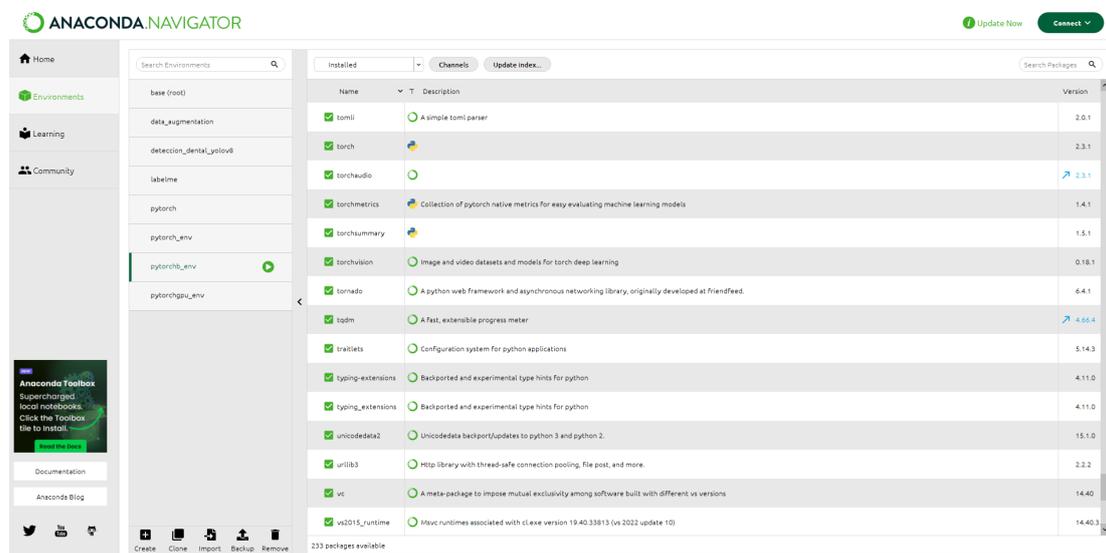


Figura 18. Entorno virtual creado - librerías

XII. MARCO METODOLÓGICO

XII-A. *Diseño General*

Ya que se busca desarrollar un sistema automatizado basado en el análisis de imágenes de rayos X, el modelo de investigación que se adoptará será exploratorio y descriptivo. Se empleará un enfoque cuantitativo para medir la precisión y eficiencia del sistema de análisis de imágenes.

XII-B. *Tipo de Investigación*

La investigación es de tipo aplicada, enfocada en resolver un problema específico en el ámbito odontológico: la detección automatizada de caries en radiografías periapicales utilizando técnicas de Deep Learning. Este enfoque busca desarrollar un modelo que mejore la precisión y eficiencia del diagnóstico clínico.

XII-C. *Nivel de Investigación*

El nivel de investigación es cuantitativo explicativo, ya que se recopilan y analizan datos numéricos para explorar la relación entre el modelo de Deep Learning y su efectividad en la detección de caries. Este enfoque permite establecer correlaciones y realizar análisis estadísticos que sustenten las conclusiones.

XII-D. *Diseño de Investigación*

El diseño es experimental pre-experimental, utilizado para manipular la variable independiente (el modelo de Deep Learning) y observar su impacto en la variable dependiente (la precisión en la detección de afecciones dentales). Este diseño es adecuado para evaluar la viabilidad del modelo en un entorno controlado.

XII-E. *Método de Investigación*

El método seguido es el hipotético-deductivo, comenzando con la formulación de hipótesis basadas en estudios previos, las cuales se verifican mediante experimentación y análisis de datos. Este enfoque permite evaluar si el modelo propuesto mejora significativamente la detección de afecciones dentales en comparación con métodos tradicionales.

XII-F. *Población y Muestra*

XII-F1. *Población:*

- Población o universo: Radiografías periapicales dentales de pacientes adultos, mayores de 17 años, obtenidas en una clínica odontológica y proporcionadas por la Dra. María Emilia Cañizares. También se obtuvieron datasets descargas de sitios web oficiales con este tipo de imágenes.
- Población de estudio: Radiografías que cumplen con criterios específicos de calidad y diagnóstico, tales como la presencia de afecciones dentales y la claridad en las imágenes.

XII-G. *Muestra*

- Unidad de análisis: Cada una de las radiografías periapicales y panorámicas utilizadas en el estudio.
- Unidad de muestreo: Las radiografías fueron etiquetadas en función de la presencia o ausencia de las afecciones dentales, proceso realizado con la colaboración de la Dra. Marcia Sacoto y el odontólogo Jordy Molina.
- Muestreo: Se empleó un muestreo intencional no aleatorio, seleccionando radiografías que cumplían con los criterios de inclusión, asegurando que las imágenes seleccionadas fueran representativas para la evaluación del modelo.

XII-H. Técnicas de Recolección de Datos

- Listas de cotejo: Utilizadas para verificar que las radiografías cumplieran con los requisitos establecidos para el estudio, como calidad de la imagen y claridad en la identificación de afecciones dentales.

XII-I. Herramientas y software:

Anaconda para gestionar entornos de desarrollo, Jupyter Notebook para la escritura y ejecución del código, y PyTorch para la creación y entrenamiento del modelo de Deep Learning. Además, se utilizó una GPU NVIDIA GeForce RTX 3050 para acelerar el proceso de entrenamiento.

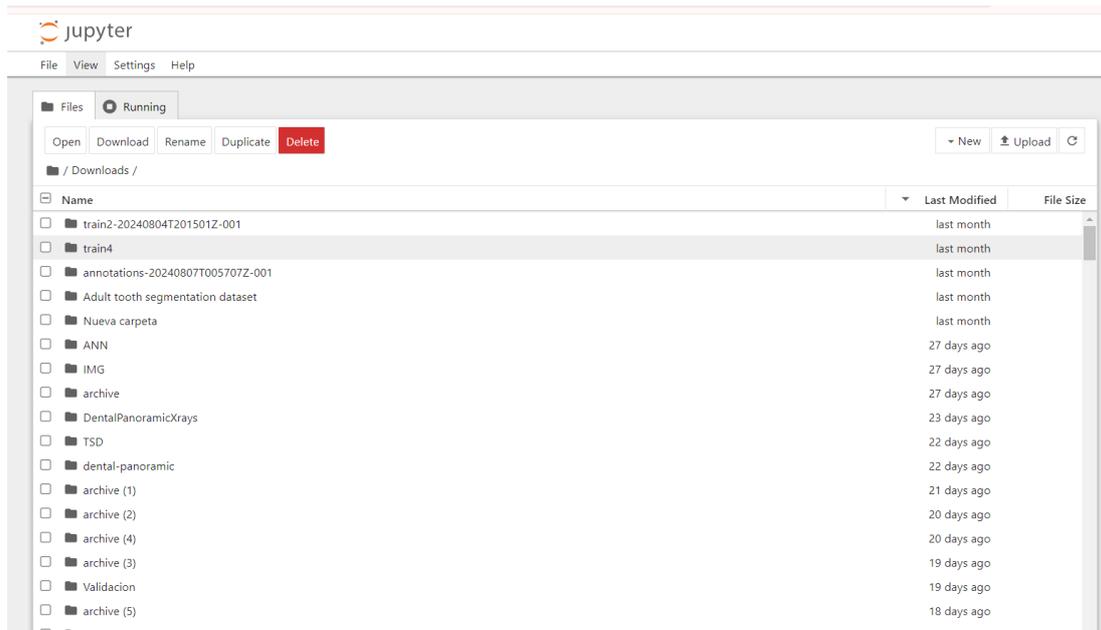


Figura 19. Entorno para ejecutar código - Jupyter

XIII. PROCEDIMIENTO METODOLÓGICO

XIV. ANÁLISIS DE DATOS

- Se realiza un análisis detallado de la precisión y la pérdida del modelo utilizando las gráficas generadas durante el entrenamiento.
- La precisión se evalúa en función de la capacidad del modelo para detectar correctamente las afecciones dentales en el conjunto de validación.

XIV-A. Implementación del modelo.

1. Definición de la métrica de precisión (Mean Average Precision)

Se inicializa la métrica Mean Average Precision (mAP) a través de 'torchmetrics', con el objetivo de calcular la precisión de detección de afecciones dentales en cada clase. Esta métrica es crucial para evaluar el rendimiento del modelo en términos de cuán correctamente puede detectar las mismas en las radiografías.

```
metric = torchmetrics.detection.MeanAveragePrecision(class_metrics=True)
```

2. Validación del modelo por época.

La función `validate_one_epoch` se encarga de validar el rendimiento del modelo después de cada época de entrenamiento. La función realiza las siguientes acciones:

- Cambia el modelo al modo de evaluación para desactivar ciertas operaciones que solo se deben realizar durante el entrenamiento, como el dropout.
- Calcula la métrica de precisión (mAP) utilizando los resultados del modelo en el conjunto de validación, comparando las predicciones con las etiquetas reales.
- Muestra los resultados de la precisión por clase, facilitando la identificación de posibles áreas de mejora en la detección.

```
def validate_one_epoch(model, loader, device):
    model.eval()
    metric.reset()

    with torch.no_grad():
        for images, targets in tqdm(loader):
            images = list(image.to(device) for image in images)
            targets = [{k: v.to(device) for k, v in t.items()}
                       for t in targets]
            outputs = model(images)

            for output in outputs:
                output['boxes'] = output['boxes'].to(device)
                output['scores'] = output['scores'].to(device)
                output['labels'] = output['labels'].to(device)

            targets = [{k: v.to(device) for k, v in t.items()}
                       for t in targets]
            metric.update(outputs, targets)

    results = metric.compute()

    if 'map_per_class' in results:
        map_per_class = results['map_per_class']
        if map_per_class.ndim > 0:
            for i, ap in enumerate(map_per_class):
                print(f"Class {i + 1}: mAP = {ap:.4f}")
        else:
            print(f"mAP for the only class: {map_per_class.item():.4f}")
    else:
        if 'map_50_per_class' in results:
            map_50_per_class = results['map_50_per_class']
            if map_50_per_class.ndim > 0:
                for i, ap in enumerate(map_50_per_class):
                    print(f"Class {i + 1}: mAP@50 = {ap:.4f}")
            else:
                print(f"mAP@50 for the only class:
```

```

        {map_50_per_class.item():.4f}")
    else:
        print("No se encontraron métricas por clase en los resultados.")

    return results

```

3. Cálculo de la precisión de predicción.

La función `calculate_accuracy` compara las predicciones del modelo con las etiquetas reales y calcula la precisión global. Este cálculo se utiliza para obtener una visión general de cuán efectivo es el modelo en términos de precisión.

4. Carga y procesamiento de anotaciones.

La función `load_labelme_annotation` carga las anotaciones de las imágenes desde archivos JSON y extrae las coordenadas de las afecciones dentales (cajas delimitadoras) y sus etiquetas. Esta función convierte los datos de anotación en un formato que puede ser utilizado por el modelo durante el entrenamiento y la validación.

```

def load_labelme_annotation(json_path, class_mapping):
    with open(json_path) as f:
        data = json.load(f)

    boxes, labels = [], []
    for shape in data['shapes']:
        points = shape['points']
        xmin = min(points[0][0], points[1][0])
        ymin = min(points[0][1], points[1][1])
        xmax = max(points[0][0], points[1][0])
        ymax = max(points[0][1], points[1][1])

        label_name = shape['label']
        if label_name in class_mapping:
            label = class_mapping[label_name]
        else:
            continue # Ignora si no está en el mapeo

        if xmax > xmin and ymax > ymin:
            boxes.append([xmin, ymin, xmax, ymax])
            labels.append(label)

    return boxes, labels

```

5. Creación del dataset personalizado.

Se define una clase `LabelmeDataset`, que extiende la funcionalidad de PyTorch para manejar un dataset personalizado.

- Carga las imágenes y las anotaciones desde las rutas especificadas.
- Aplica transformaciones a las imágenes para prepararlas para el modelo.
- Devuelve las imágenes y las etiquetas correspondientes en un formato adecuado para el entrenamiento del modelo.

```

class LabelmeDataset(torch.utils.data.Dataset):
def __init__(self, img_dir, annotation_dir
class_mapping, transform=None):
    self.img_dir = img_dir
    self.annotation_dir = annotation_dir
    self.class_mapping = class_mapping
    self.transform = transform
    self.img_ids = [f.split('.')[0]
for f in os.listdir(img_dir) if f.endswith('.jpg')]

def __len__(self):
    return len(self.img_ids)

def __getitem__(self, idx):
    img_id = self.img_ids[idx]
    img_path = os.path.join(self.img_dir, img_id + '.jpg')
    ann_path = os.path.join(self.annotation_dir, img_id + '.json')

    image = Image.open(img_path).convert("RGB")
    boxes, labels = load_labelme_annotation(ann_path, self.class_mapping)

    target = {'boxes': torch.tensor(boxes, dtype=torch.float32),
              'labels': torch.tensor(labels, dtype=torch.int64)}

    if self.transform:
        image = self.transform(image)

    return image, target

```

XV. ENTRENAMIENTOS

XV-A. Selección del entorno de trabajo y herramientas

El proyecto se comenzó utilizando Google Colab como entorno de desarrollo, aprovechando su infraestructura de GPU para entrenar modelos de redes neuronales convolucionales (CNN) de manera remota. Sin embargo, debido a las limitaciones inherentes de este entorno, como la inestabilidad y la desconexión automática de las sesiones cuando los entrenamientos superaban las 2 horas, el entrenamiento de los modelos resultaba ineficiente. Se observó que los tiempos de entrenamiento oscilaban entre 12 y 33 horas, lo que hacía inviable la obtención de resultados óptimos dentro de los plazos requeridos para el desarrollo del sistema.

The screenshot displays a Google Colab interface. On the left, a code cell contains the following Python code:

```
plt.title('Training Loss')
plt.legend()

# Precisión de entrenamiento
plt.subplot(1, 2, 2)
plt.plot(train_accuracies, label='Train Accuracy')
plt.xlabel('Epoch')
plt.ylabel('Accuracy')
plt.title('Training Accuracy')
plt.legend()

plt.tight_layout()
plt.show()
```

Below the code, the terminal output shows a warning and a download progress bar:

```
/usr/local/lib/python3.10/dist-packages/torchvision/models/_utils.py:208: UserWarning: The par
warnings.warn(
/usr/local/lib/python3.10/dist-packages/torchvision/models/_utils.py:223: UserWarning: Argumer
warnings.warn(msg)
Downloading: "https://download.pytorch.org/models/fasterrcnn_resnet50_fpn_coco-258fb6c6.pth" t
100%|██████████| 160M/160M [00:01<00:00, 107MB/s]
Epoch 0, Loss: 0.08322630177896756, Accuracy: 0.0
```

On the right side, the 'Recursos' (Resources) panel shows system status:

- RAM: 1.2 / 12.7 GB
- Disco: 31.0 / 107.7 GB

There are also several informational messages and a notification about upgrading to Colab Pro.

Figura 20. Entorno de entrenamiento por C. Franco, google colab.

Se puede observar como en google colab los tiempos de entrenamiento eran demasiado largos y poco viables.

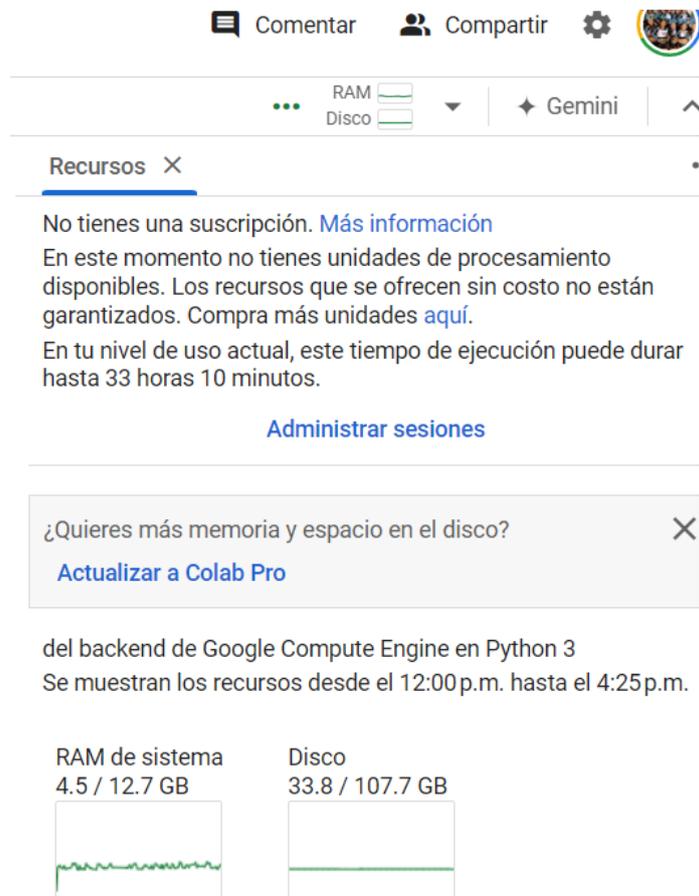


Figura 21. Entorno de entrenamiento 2 por C. Franco, google colab.

En vista de estos inconvenientes, se decidió migrar el entorno de trabajo a una laptop con GPU NVIDIA GeForce RTX 3050 y 16 GB de RAM. El entorno fue configurado utilizando Anaconda como gestor de paquetes y PyTorch como la biblioteca principal para la implementación de modelos de redes neuronales. También se empleó Jupyter Notebook para una visualización más clara y manejable del código, y *LabelMe* para la anotación manual de las imágenes de rayos X panorámicas y periapicales.

Este cambio permitió un control más preciso sobre el hardware y la reducción de los tiempos de entrenamiento, aunque no eliminó completamente los desafíos inherentes al procesamiento de imágenes médicas.

XV-B. Dataset utilizado

El conjunto de datos inicial constaba de 600 imágenes de rayos X panorámicos, 100 de ellas se anotaron manualmente utilizando el formato JSON provisto por la herramienta *LabelMe*. Las etiquetas iniciales incluían cuatro afecciones dentales: caries, lesión periapical, diente impactado e implante. Estas imágenes fueron seleccionadas debido a su relevancia clínica en la práctica odontológica y su prevalencia en las radiografías dentales. Finalmente se logró obtener un dataset de 5000 imágenes, entre ellas imágenes de rayos x panorámicas y periapicales.

XV-C. Modelo utilizado

El modelo seleccionado para la tarea fue una red *Faster R-CNN* basada en *ResNet-50* con *Feature Pyramid Networks (FPN)*, un enfoque ampliamente utilizado en tareas de detección de objetos debido a su capacidad para extraer características de diferentes escalas y manejar la variabilidad en el tamaño de los objetos detectados. Se empleó una versión preentrenada del modelo, utilizando pesos inicializados en un conjunto de datos grande y genérico (*COCO dataset*), lo que proporcionó una ventaja en términos de aprendizaje de características básicas de las imágenes.

El número de clases se definió finalmente como 5: caries(clase 1), diente impactado (clase 2), lesión periapical(clase 3), implante (clase 4) y una clase de fondo (clase 0). Es importante incluir la clase de fondo en modelos de detección de objetos para que el modelo aprenda a distinguir entre las regiones de interés y aquellas sin relevancia.

XV-D. Procedimiento de entrenamiento

El entrenamiento del modelo se realizó con un enfoque incremental. En una primera instancia se trabajó con imágenes de rayos x panorámicas, se entrenó utilizando 100 imágenes, luego 200 y finalmente 846, a fin de evaluar el comportamiento del modelo con diferentes tamaños de datos. Sin embargo, debido a la complejidad de las clases de caries y lesión periapical, el modelo mostró una incapacidad para aprender de manera efectiva estas afecciones. Los resultados preliminares indicaron un aprendizaje casi nulo con estas clases, con precisiones extremadamente bajas, lo que llevó a la decisión de continuar solo con diente impactado e implante como clases objetivo.

- Class 1 : Caries.
- Class 2: diente impactado.
- Class 3: lesión periapical.
- Class 4: implante.

```
Class 1: mAP = 0.0023
Class 2: mAP = 0.1295
Class 3: mAP = 0.0000
Class 4: mAP = 0.3894
Validation mAP (Average): 0.1303
mAP@50: 0.2746
mAP@75: 0.1198
```

Figura 22. Entrenamientos previos, por C. Franco Jupyter notebook.

Se realizaron posteriores entrenamientos, utilizando más dataset y aumento de datos, sin embargo los resultados seguían teniendo la misma inclinación de aprendizaje, se concluye que la red no puede aprender mediante imágenes panorámicas estos dos tipos de afecciones dentales.

En la fase final, el modelo fue entrenado con un conjunto de 846 imágenes, utilizando técnicas de *Data Augmentation* (transformaciones como flips horizontales y cambios de escala) para mejorar la generalización del modelo.

XV-E. Análisis de los resultados y errores iniciales.

A pesar de los esfuerzos por mejorar el rendimiento del modelo, varios factores limitaron su capacidad para generalizar adecuadamente:

- Complejidad de las imágenes: Las imágenes de rayos X dentales presentan desafíos inherentes debido a su bajo contraste, ruido y la superposición de estructuras anatómicas. Esto dificulta que el modelo identifique con precisión los contornos de las afecciones.
- Limitación del dataset: Aunque se emplearon más de 1000 imágenes en algunos entrenamientos, este número sigue siendo relativamente bajo para un problema de detección de objetos. Además, el desbalance entre las clases y la posible presencia de imágenes mal etiquetadas (cajas vacías o mal anotadas) afectó el rendimiento del modelo.
- Subentrenamiento: El número de épocas de entrenamiento fue limitado a solo 50, lo que probablemente no fue suficiente para que el modelo convergiera adecuadamente. Un mayor número de épocas podría haber permitido un ajuste más fino de los pesos del modelo, mejorando su capacidad para detectar las afecciones.
- Tasa de aprendizaje: La tasa de aprendizaje utilizada (0.005) puede haber sido demasiado alta, lo que impidió que el modelo ajustara sus pesos de manera adecuada. Al reducir la tasa de aprendizaje o implementar una política de reducción adaptativa tampoco se logró una mejora.
- Métricas (mAP): Se utilizó la métrica *Mean Average Precision (mAP)* para evaluar el desempeño del modelo en la detección de objetos. Los resultados indicaron un *mAP@50* relativamente bajo, lo que sugiere que el modelo no pudo detectar las afecciones con la suficiente precisión. Esto también refleja que los falsos positivos y falsos negativos seguían siendo un problema.

XVI. ENTRENAMIENTO DEL MODELO FASTER R-CNN: ANÁLISIS DETALLADO

XVI-A. Modelo Utilizado

- Arquitectura: Faster R-CNN
- Backbone: ResNet-50
- Framework: PyTorch

El modelo Faster R-CNN es una arquitectura avanzada para detección de objetos que combina una red de regiones propuestas (RPN) y un detector de objetos basado en una red neuronal convolucional. La red ResNet-50 sirve como el backbone, proporcionando características de alto nivel para la detección de objetos.

A continuación se mostrarán los entrenamientos que se pudieron culminar con mayor satisfacción:

XVI-B. Prueba técnica 1

- Número de Imágenes: 100
- Clases: 4 (Caries, Lesión Periapical, Diente Impactado, Implante) + Fondo

Parámetros de Entrenamiento:

- Épocas: 3
- Precisión: 0.09
- Pérdida: 0.31

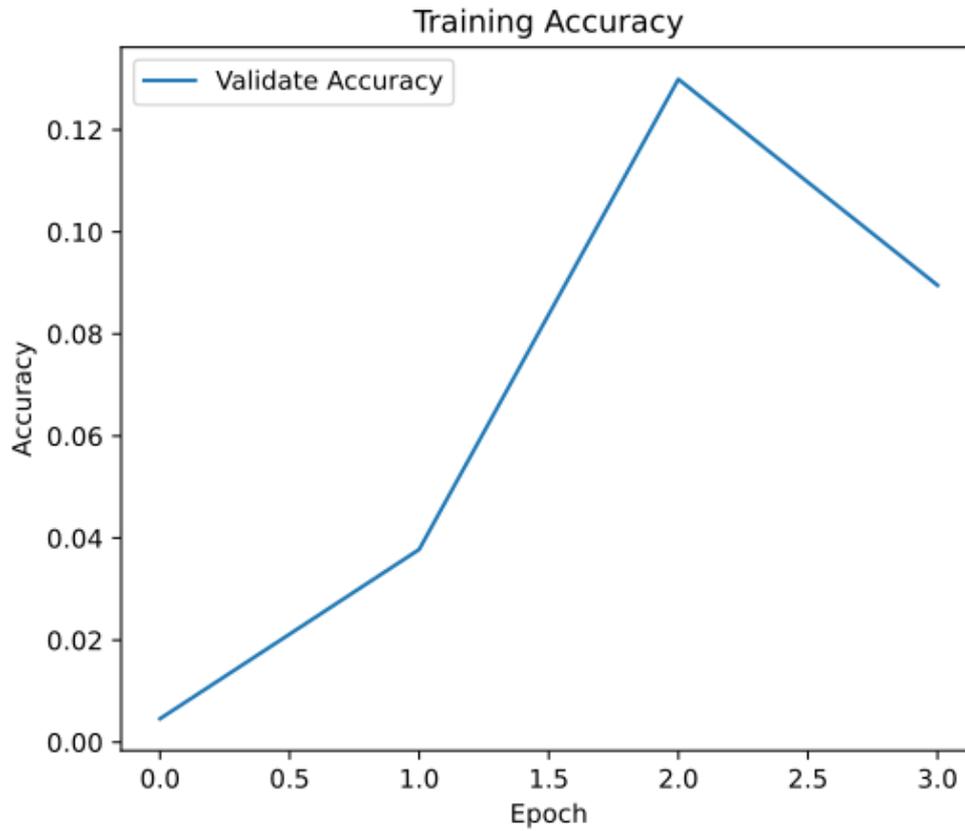


Figura 23. Resultado mAP de la prueba 1 con 4 clases más una de fondo.

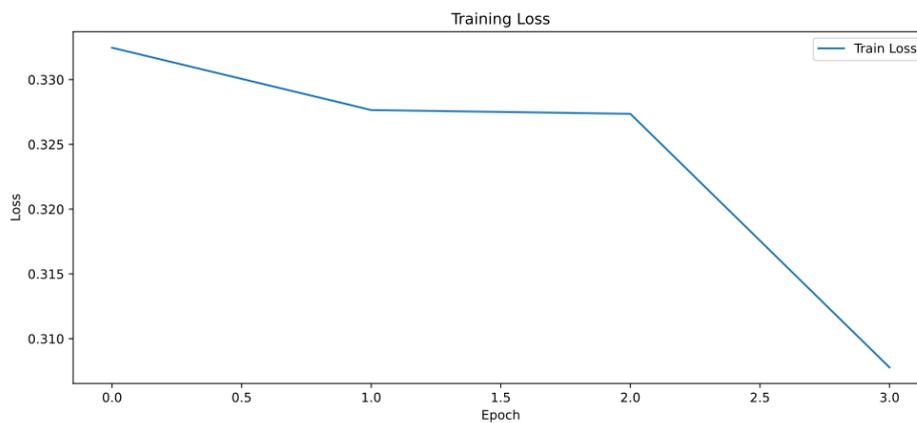


Figura 24. Pérdida durante el entrenamiento en la prueba 1.

Observaciones: El modelo mostró una baja precisión de 0.09 y una pérdida de 0.31 después de 3 épocas. La baja precisión puede indicar que el modelo no está aprendiendo patrones significativos de las imágenes o que la cantidad de épocas es insuficiente para capturar la variabilidad en los datos.

Conclusión: El bajo rendimiento inicial podría estar relacionado con el tamaño reducido del dataset o con una

configuración subóptima del modelo. Es posible que el modelo necesite más épocas para converger.

XVI-C. Prueba técnica 2

- Número de Imágenes: 100
- Épocas: 14
- Precisión Máxima: 0.16 (en la época 3)
- Precisión Promedio: 0.10
- Pérdida Final: 0.16

Observaciones: Se observó una alta variabilidad en la precisión durante el entrenamiento, con fluctuaciones en el mAP (mean Average Precision). Esto sugiere que el modelo podría estar sobreajustándose a las imágenes de entrenamiento pequeñas. La precisión promedio de 0.10 indica que el modelo aún no ha aprendido de manera efectiva a distinguir entre las clases.

Conclusión: Es probable que la falta de consistencia en el aprendizaje se deba a un tamaño de dataset insuficiente. Se recomienda aumentar el número de imágenes y ajustar los hiperparámetros.

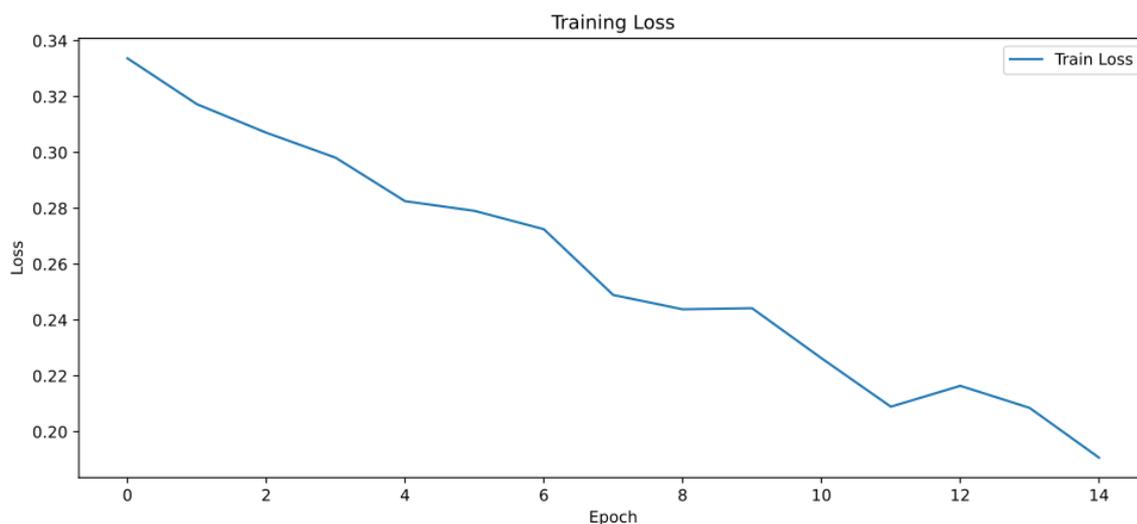


Figura 25. Curva de pérdida durante el entrenamiento de la Prueba 2

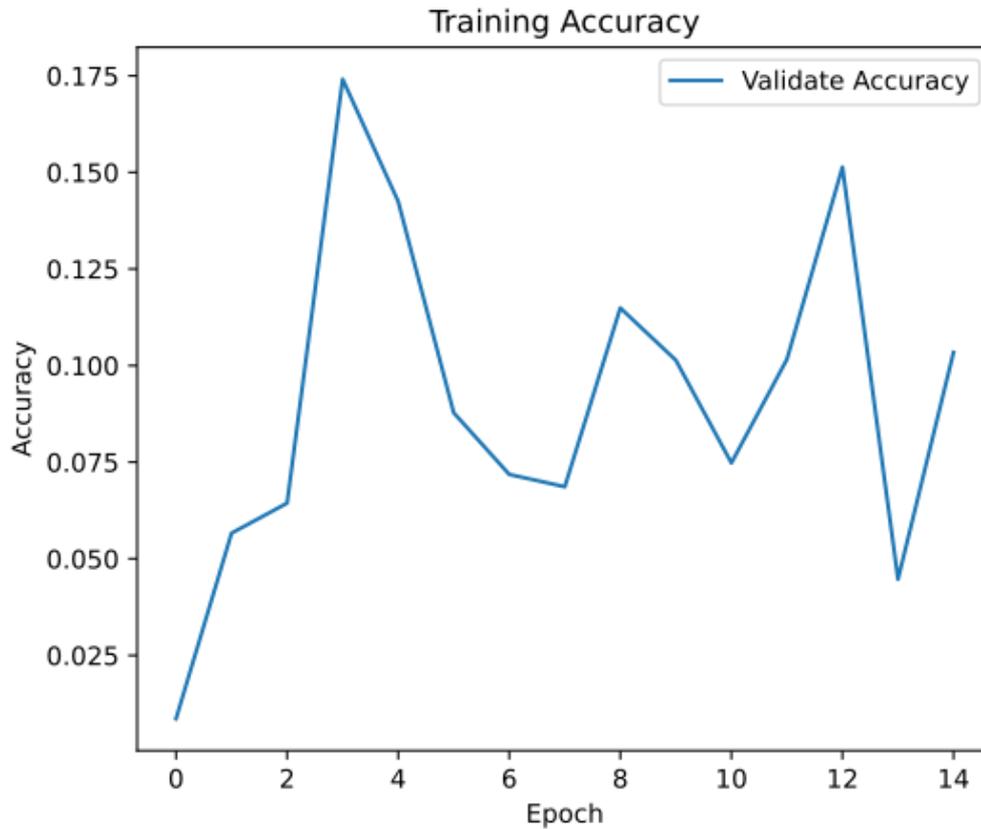


Figura 26. Curva de mAP durante el entrenamiento de la Prueba 2

XVI-D. Prueba técnica 3

- Número de Imágenes: 782
- Clases: 4
- Épocas: 12
- Precisión Promedio: 0.16

Observaciones: El incremento en el número de imágenes (782) no resultó en una mejora significativa en la precisión. Esto puede indicar problemas con la calidad de las anotaciones o con el balance entre clases.

Conclusión: Aunque se incrementó el tamaño del dataset, la precisión no mejoró sustancialmente, lo cual puede deberse a que el modelo no está generalizando bien. La estrategia de etiquetado puede necesitar ajustes.

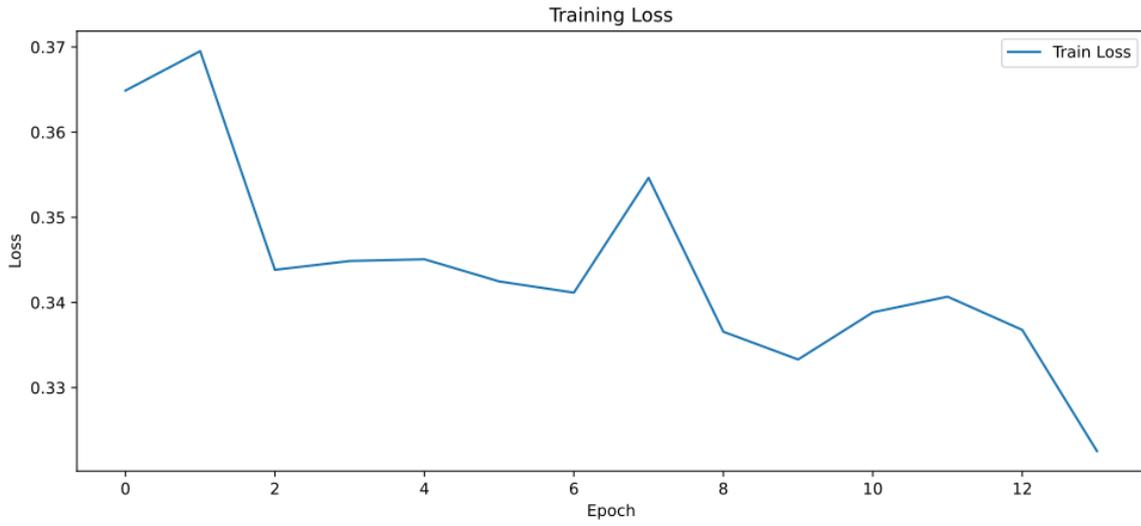


Figura 27. Curva de pérdida durante el entrenamiento de la Prueba 3

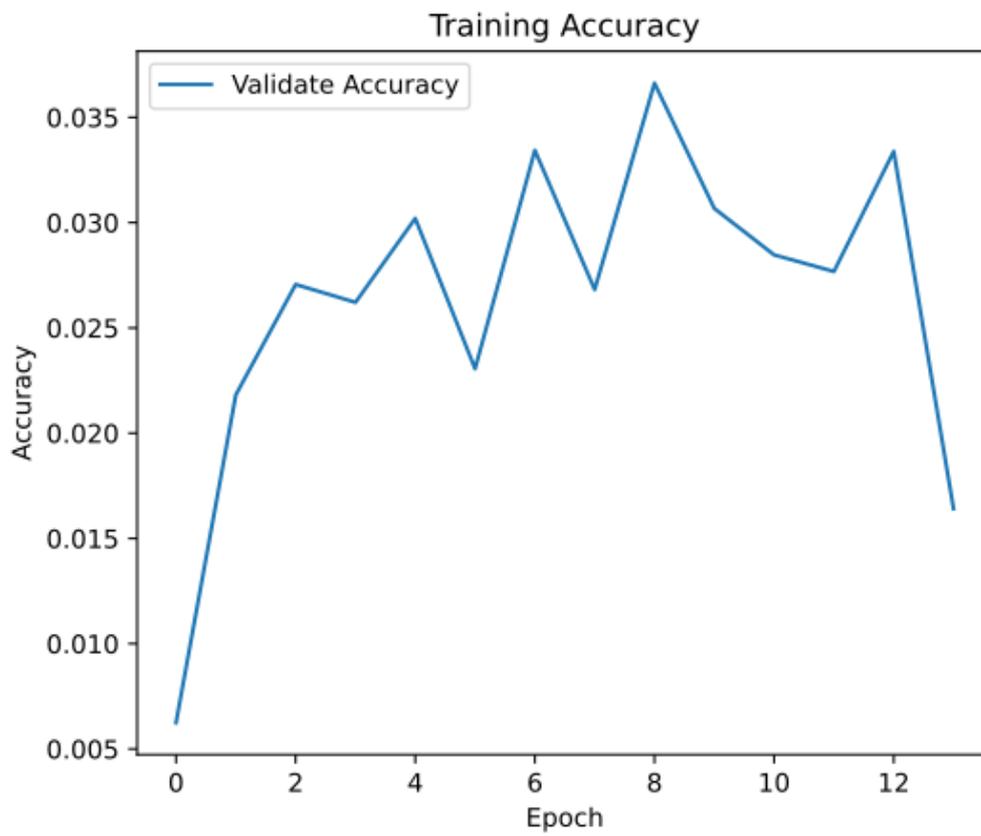


Figura 28. Curva de mAP durante el entrenamiento de la Prueba 3

XVI-E. Prueba técnica 4

- Número de Imágenes: 832 (782 iniciales + 50 adicionales)
- Épocas: 5
- Precisión Promedio: 0.175
- Pérdida: 0.27

Observaciones: La precisión mejoró ligeramente con la adición de 50 imágenes, pero no de manera significativa. El tiempo de entrenamiento se redujo, lo cual podría indicar que el modelo no está convergiendo adecuadamente.

Conclusión: La precisión mejoró ligeramente, pero el aumento de datos no ha llevado a una mejora significativa. Se procedió a revisar el proceso de anotación y aumentar aún más el número de épocas.

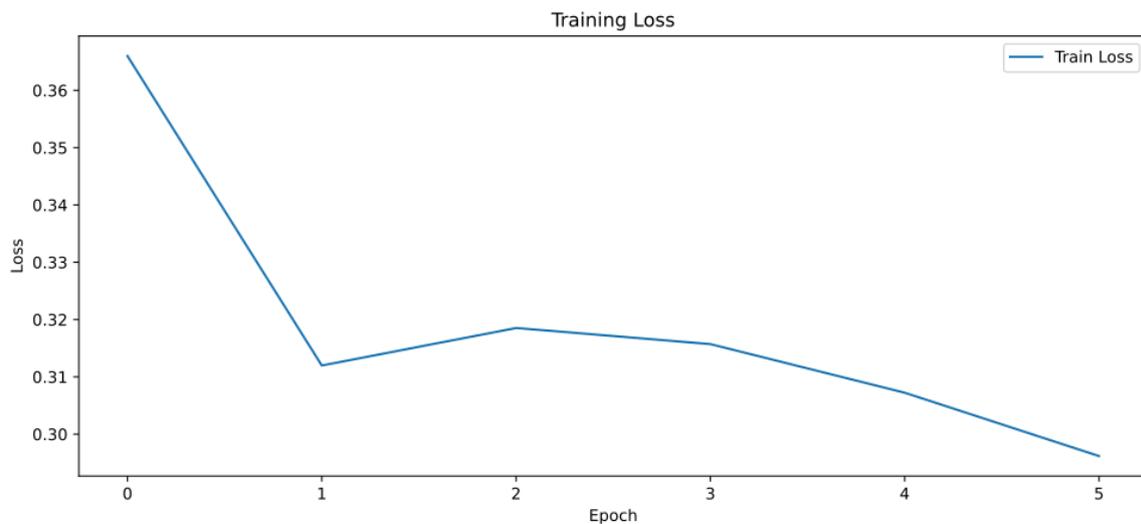


Figura 29. Curva de pérdida durante el entrenamiento de la Prueba 4

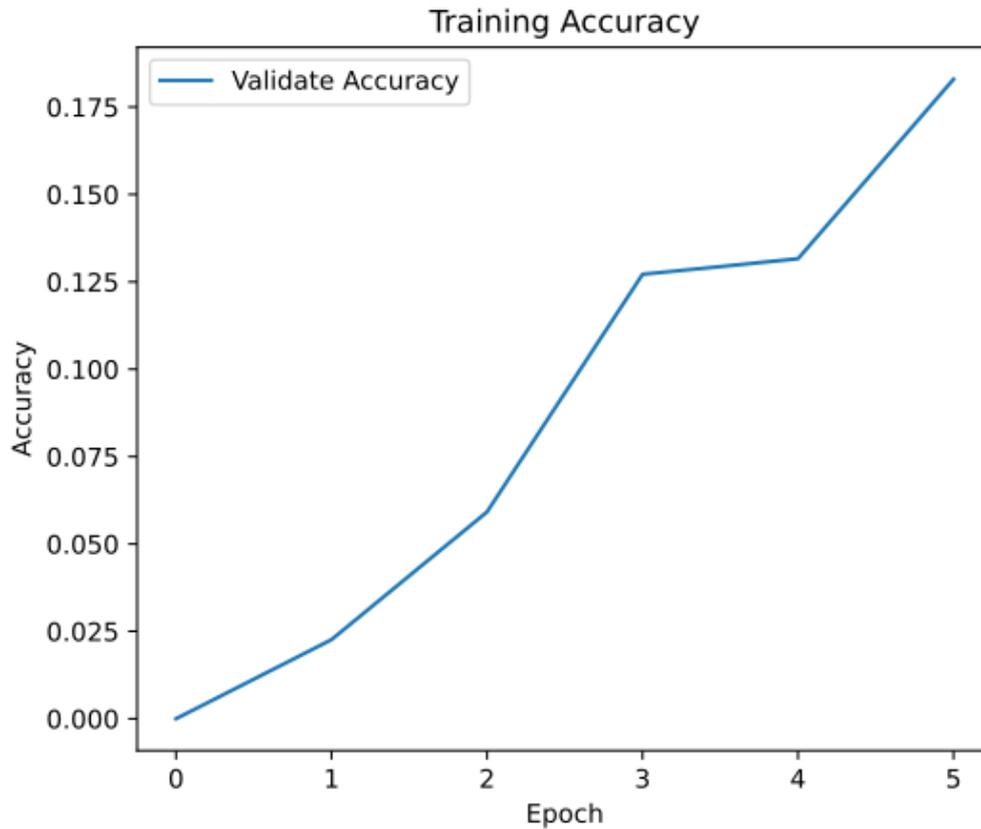


Figura 30. Curva de mAP durante el entrenamiento de la Prueba 4

XVI-F. Prueba técnica 5

- Número de Imágenes: 782
- Método de Etiquetado: Poligonal (dientes individuales)
- Épocas: 18
- Precisión Promedio: 0.05

Observaciones: El cambio en el método de etiquetado a una forma más detallada (poligonal) mostró un aprendizaje aún más complicado (0.050), aunque la red todavía no aprende de manera óptima. Esto puede indicar que la precisión es más dependiente del método de etiquetado y calidad de imágenes.

Conclusión: El etiquetado poligonal no ofreció una mejora, no fue suficiente para una detección robusta.

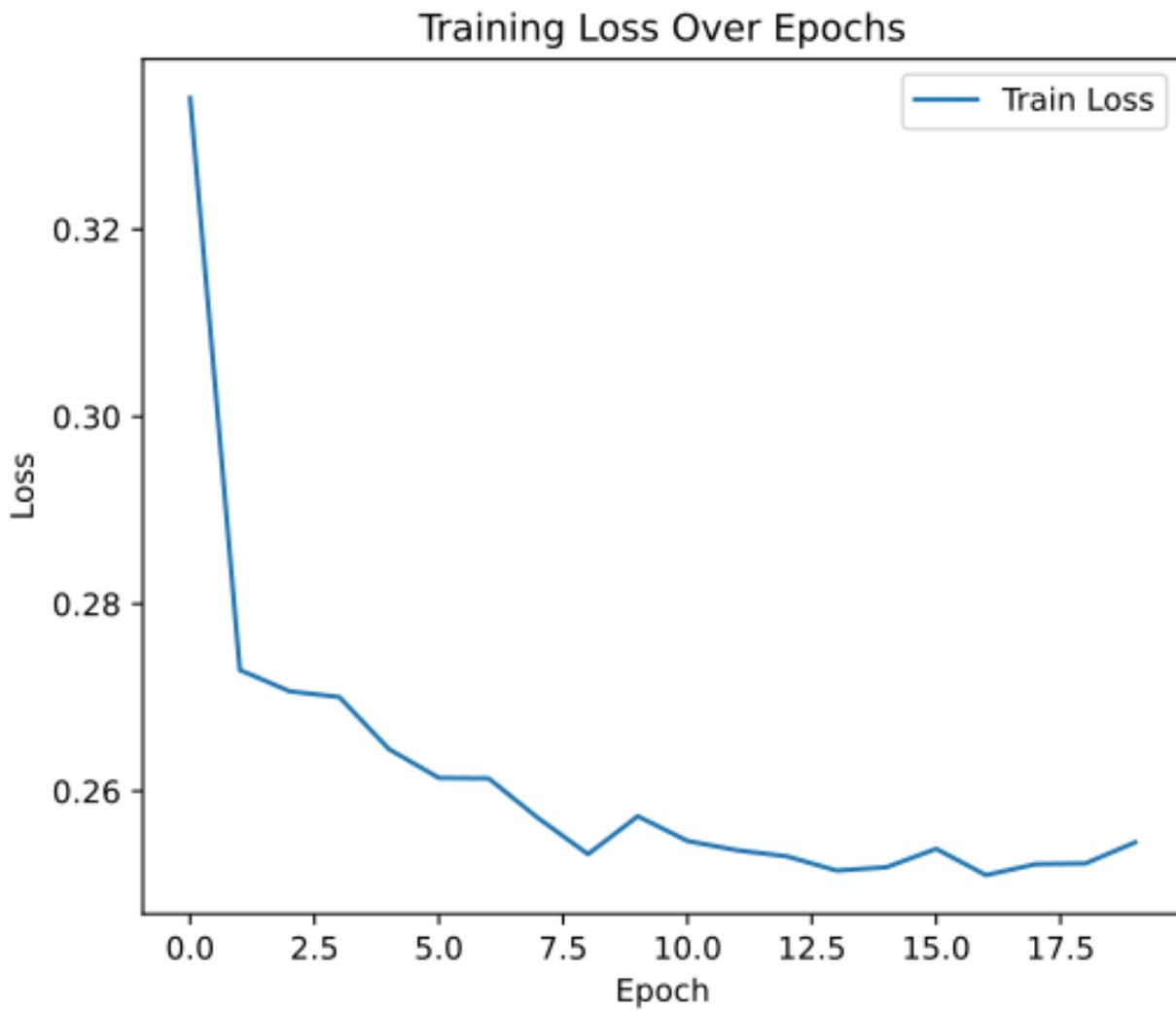


Figura 31. Curva de pérdida durante el entrenamiento de la Prueba 5

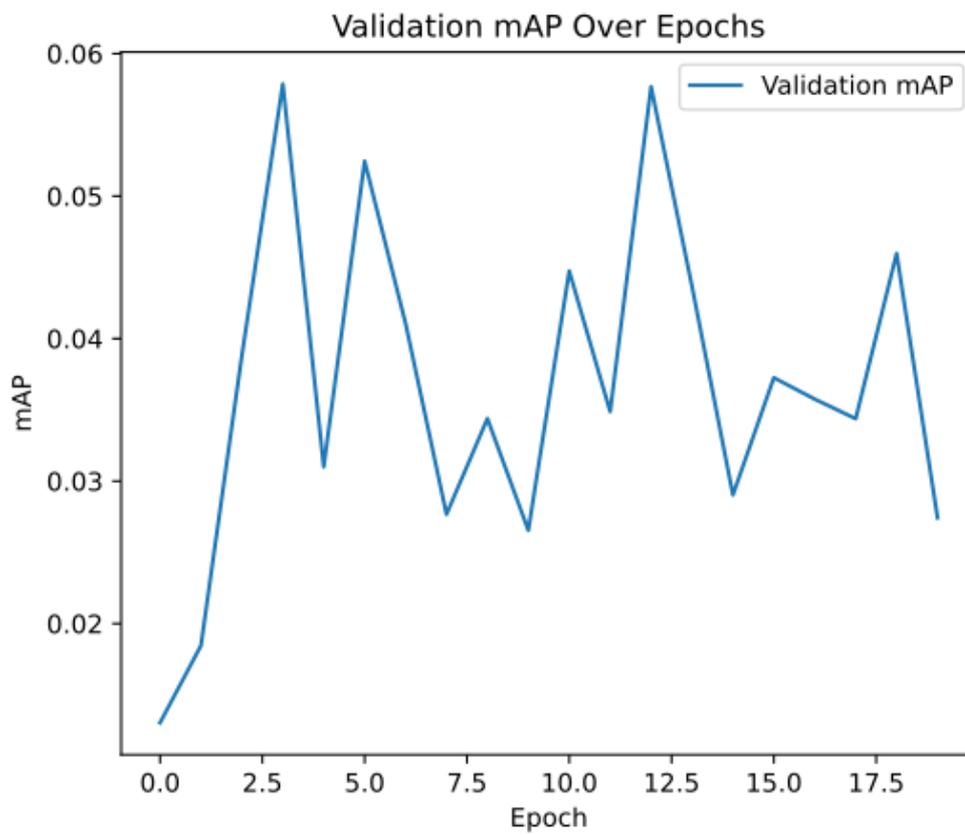


Figura 32. Curva de mAP durante el entrenamiento de la Prueba 5

XVI-G. Prueba técnica 6

- Número de Imágenes: 782
- Método de Etiquetado: Rectangular.
- Épocas: 25
- Precisión: 0.175

Observaciones: El entrenamiento con 782 imágenes panorámicas mejoró la precisión a 0.175. El aumento en la cantidad de épocas parece tener un efecto positivo en el aprendizaje del modelo.

Conclusión: Aumentar el número de épocas ha mostrado resultados prometedores. Sin embargo, la precisión sigue siendo baja, lo que sugiere que aún se necesita más ajuste y optimización.

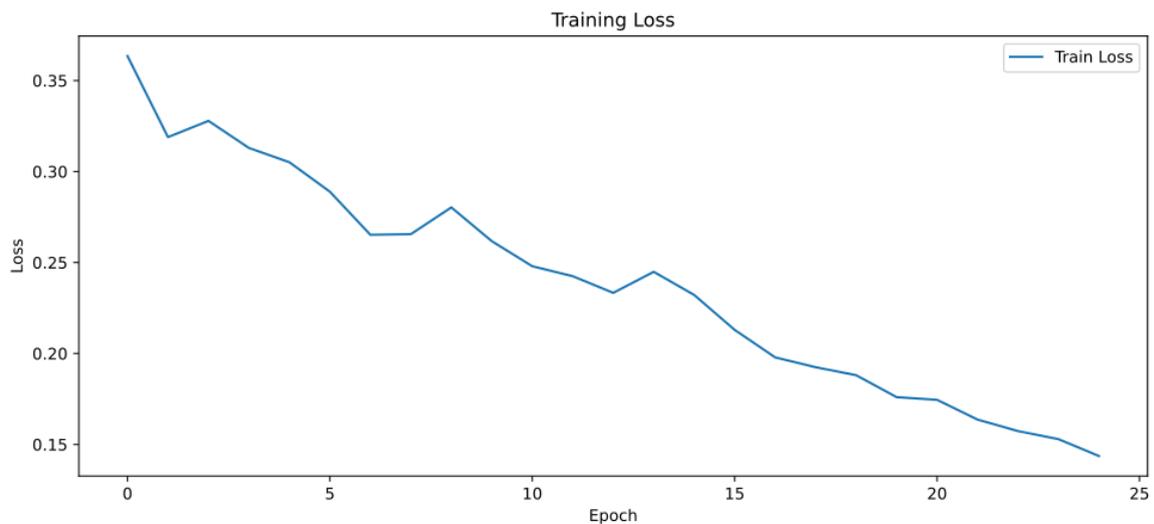


Figura 33. Curva de pérdida durante el entrenamiento de la Prueba 6

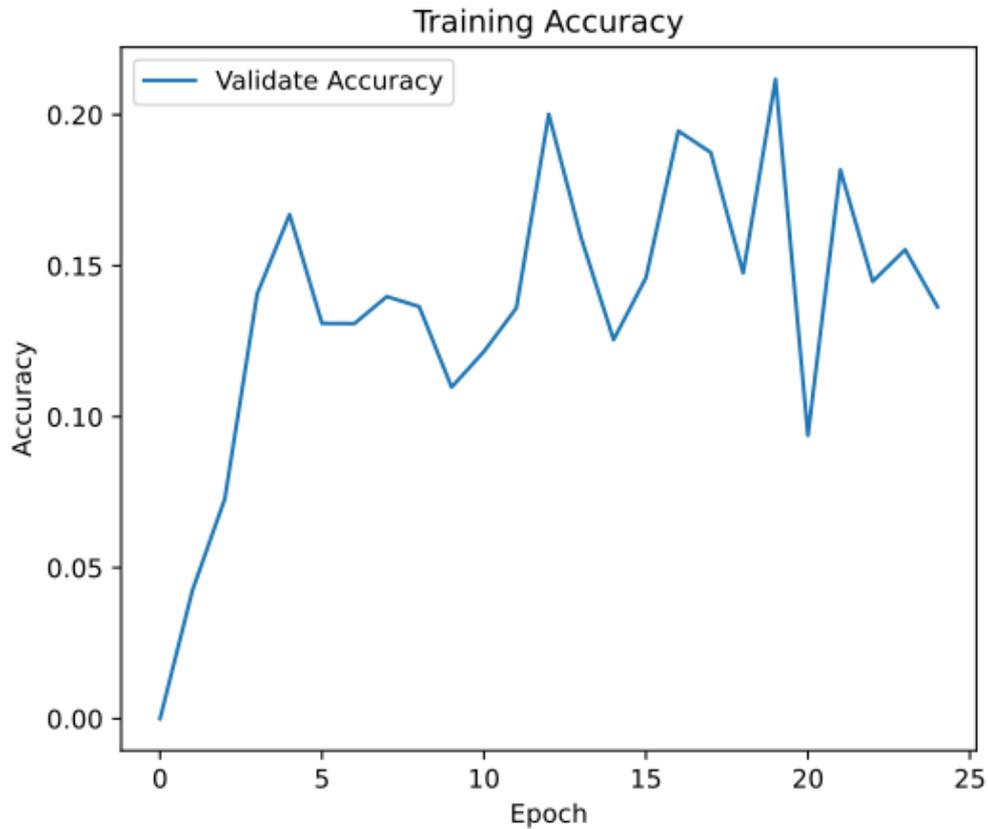


Figura 34. Curva de mAP durante el entrenamiento de la Prueba 6

XVI-H. Prueba técnica 7

- Número de Imágenes: 400 (periapicales de caries)
- Épocas: 30
- Precisión: 0.37

Observaciones: El entrenamiento con 400 imágenes periapicales mejoró la precisión a 0.37. El aumento en la cantidad de datos periapicales parece tener un efecto positivo en el aprendizaje del modelo.

Conclusión: El uso de imágenes periapicales ha mostrado resultados prometedores. Sin embargo, la precisión sigue siendo baja, lo que sugiere que aún se necesita más ajuste y optimización.

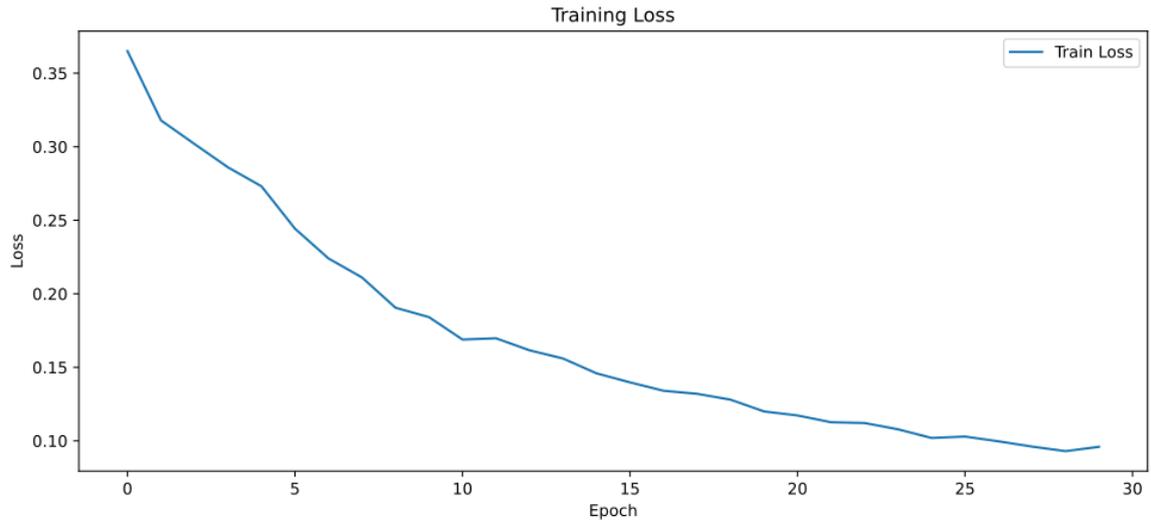


Figura 35. Curva de pérdida durante el entrenamiento de la Prueba 7

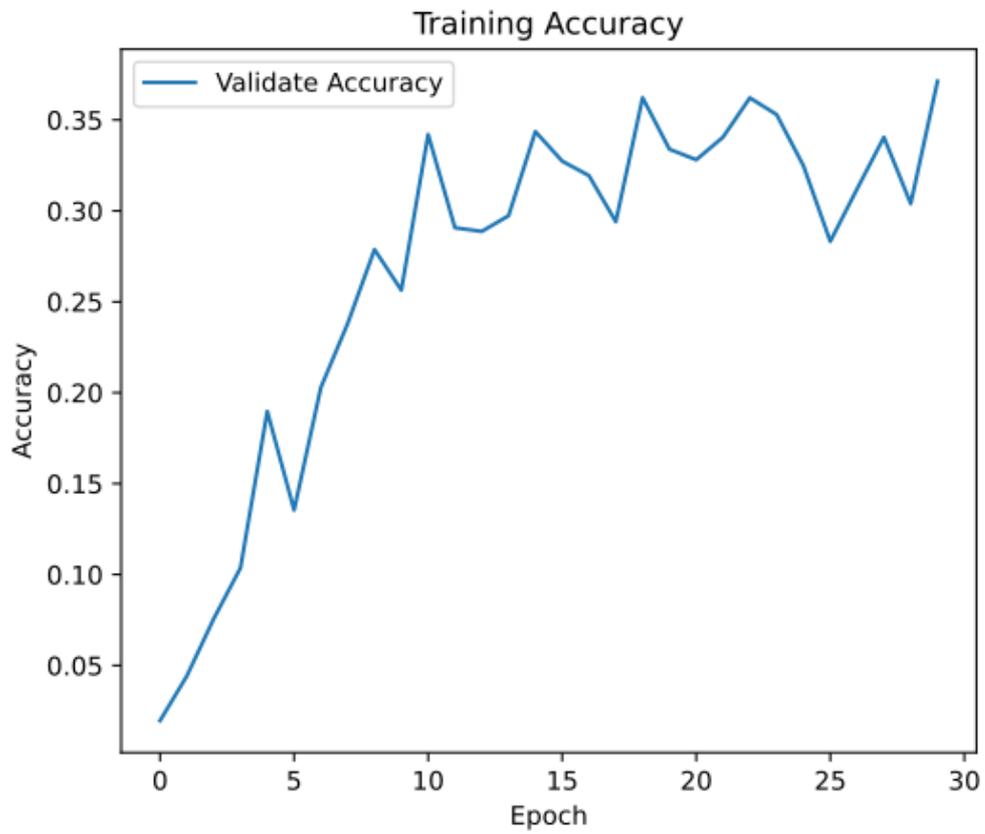


Figura 36. Curva de mAP durante el entrenamiento de la Prueba 7

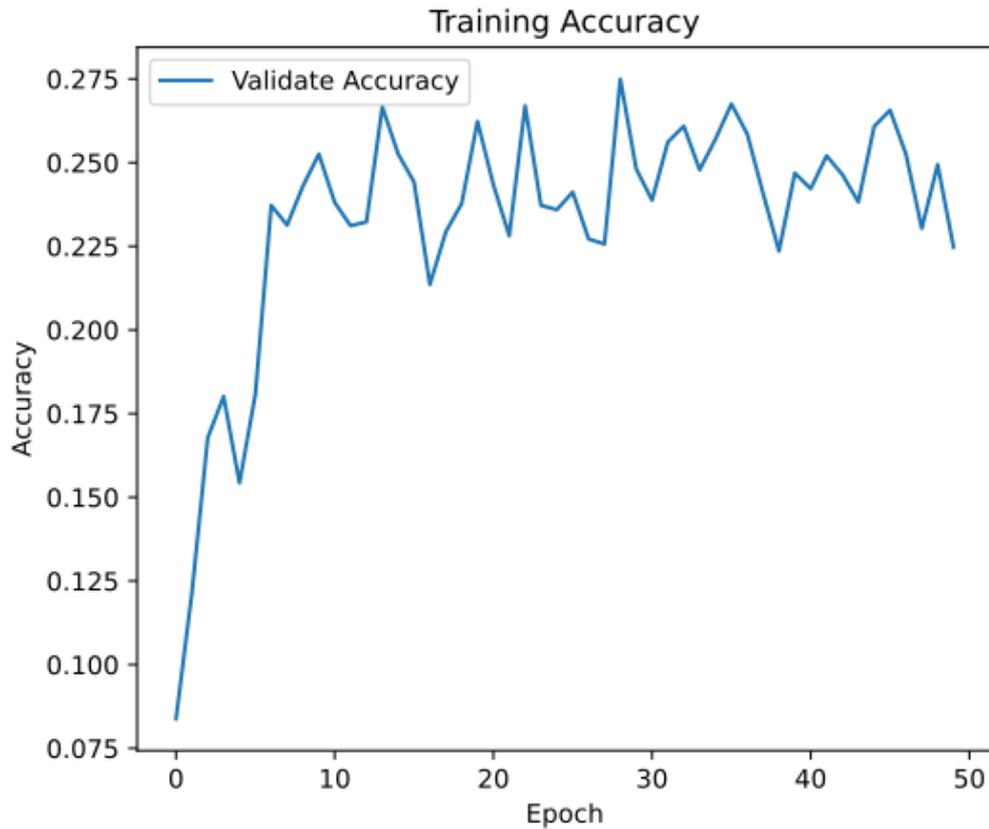


Figura 39. Curva de mAP durante el entrenamiento de la Prueba 8

XVI-J. Prueba técnica 9

- Número de Imágenes: 1068 (se realizó aumento de datos con transformaciones en imágenes panorámicas).
- Clases: 2 Diente Impactado e Implante
- Épocas: 14
- Precisión: 0.20

Observaciones: De acuerdo a recomendaciones del tutor se toma como objetivo las afecciones dentales diente impactado e implante. Se seleccionaron las imágenes con este tipo de afección y se procedió a realizar aumento de datos con transformaciones no resultó en una mejora significativa con respecto a entrenamientos anteriores que incluían esta afección. La precisión observada es 0.20, similar a pruebas anteriores.

Conclusión: Las técnicas de aumento de datos parecen no haber mejorado considerablemente la precisión.

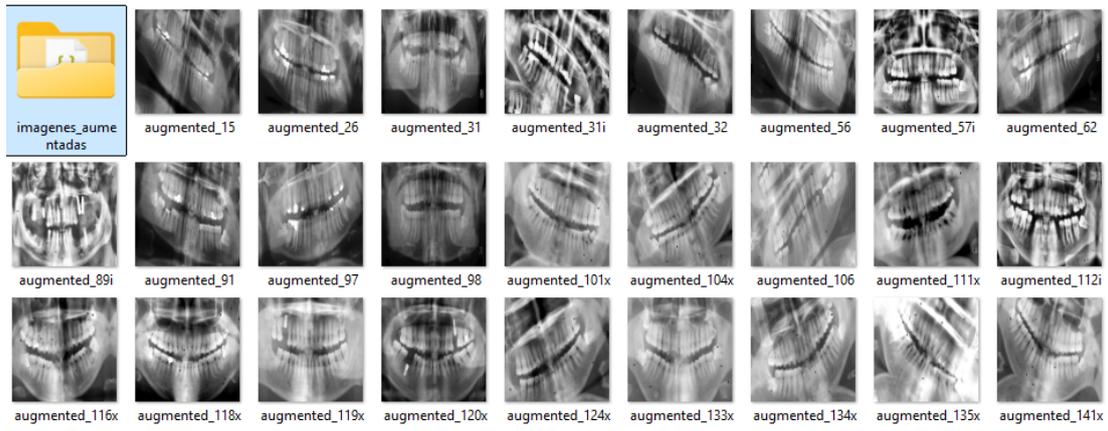


Figura 40. Imágenes panorámicas aumentadas en prueba 9

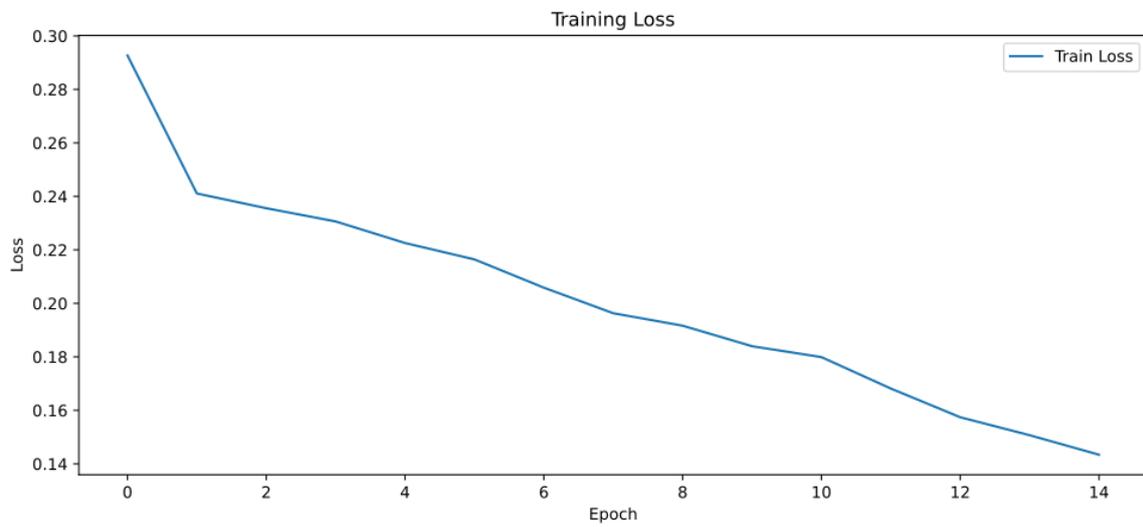


Figura 41. Curva de pérdida durante el entrenamiento de la Prueba Final

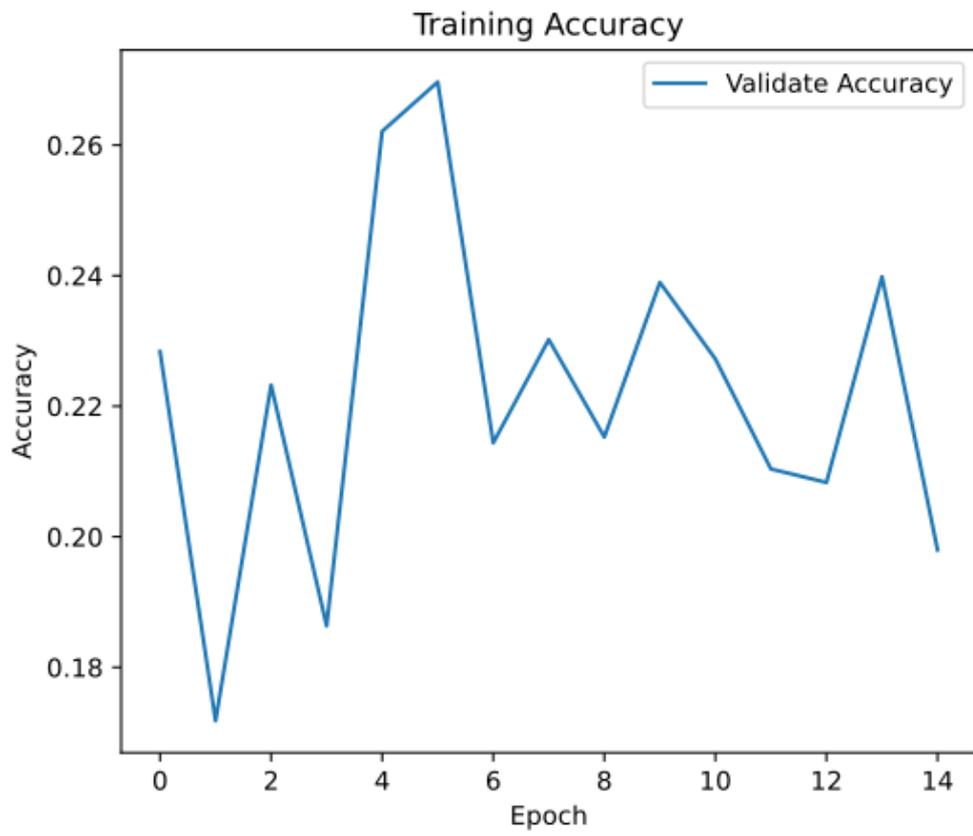


Figura 42. Curva de mAP durante el entrenamiento de la Prueba 9

XVI-K. Prueba técnica final

- Número de Imágenes: 846 (se se seleccionó las imágenes panorámicas con mayor resolución).
- Clases: 2 Diente Impactado e Implante
- Épocas: 6
- Precisión: 0.23

Observaciones: La selección de imágenes de mayor resolución y claridad ha dado lugar a una ligera mejora en la precisión, alcanzando 0.23.

Conclusión: Las imágenes de mayor calidad y resolución parecen tener un impacto positivo en la precisión del modelo. Sin embargo, la precisión sigue siendo baja, lo que indica que el modelo aún tiene espacio para mejorar.

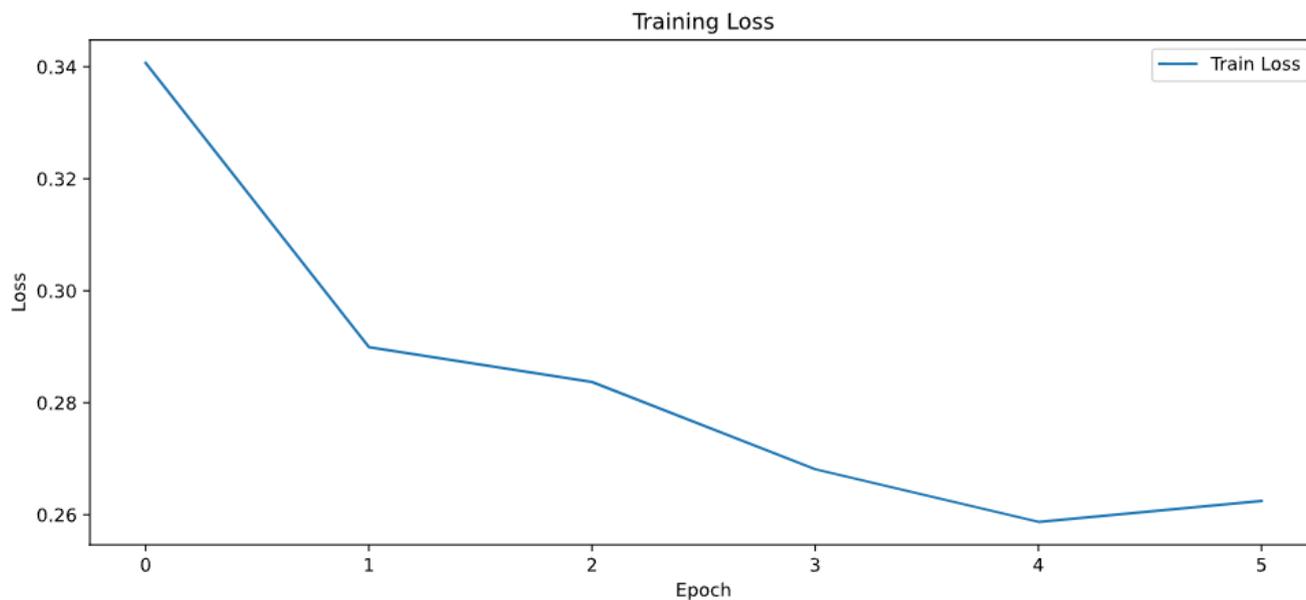


Figura 43. Curva de pérdida durante el entrenamiento de la Prueba final

XVII. RESULTADOS

El modelo *Faster R-CNN* mostró un rendimiento alentador en la detección de afecciones dentales, específicamente en las clases de *diente impactado* e *implantes dentales*, lo que evidencia su capacidad para aprender características relevantes a partir de imágenes de rayos X panorámicas. A pesar de que el modelo no alcanzó la precisión óptima, los resultados obtenidos son un indicativo positivo de que el sistema está adquiriendo conocimiento y puede mejorar significativamente con ajustes futuros.

XVII-A. Precisión y Aprendizaje del Modelo

El modelo alcanzó un *Mean Average Precision (mAP)* máximo de 0.23, lo que sugiere que, aunque no llegó a los niveles de precisión ideales para aplicaciones clínicas, sí logró aprender patrones útiles en las imágenes. Este nivel de precisión es prometedor, ya que refleja que el modelo está detectando correctamente las clases objetivo en un número considerable de casos. La consistencia del aprendizaje observada a través de varias pruebas indica que el *Faster R-CNN* está desarrollando una comprensión progresiva de las características esenciales de los *dientes impactados* e *implantes dentales*.

XVII-B. Impacto del Aumento de Datos (*Data Augmentation*)

Se implementaron técnicas de *Data Augmentation* como reflejos horizontales, rotaciones y cambios de escala, con el objetivo de aumentar la diversidad del conjunto de datos y mejorar la robustez del modelo. Aunque las mejoras en precisión no fueron significativas, estas técnicas contribuyeron a que el modelo pudiera generalizar mejor a partir de un conjunto de datos limitado, lo que sugiere que un aumento en la cantidad y diversidad de imágenes podría llevar a mejoras más notables.

XVII-C. Oportunidades de Mejora

El tamaño del dataset, compuesto por 912 imágenes de rayos X, es una limitación que debe abordarse en futuros trabajos. Un conjunto de datos más grande y diverso permitiría que el modelo entrenara con mayor cantidad de ejemplos, lo cual es fundamental para mejorar su precisión en la detección de afecciones dentales. A pesar de esta limitación, el modelo mostró un comportamiento estable durante el entrenamiento, lo que indica que con más imágenes, podría aumentar su capacidad para generalizar y mejorar su rendimiento.

XVII-D. Perspectiva Futura

Aunque la precisión actual del modelo podría considerarse baja para un entorno clínico, su capacidad de aprendizaje ya demuestra potencial para ser optimizado. El uso de un conjunto de datos más extenso y mejor anotado, junto con ajustes en los hiperparámetros, como la tasa de aprendizaje, podría potenciar aún más el rendimiento del *Faster R-CNN*. Además, la implementación de técnicas de preprocesamiento más avanzadas o la inclusión de imágenes más variadas podrían llevar al modelo a alcanzar niveles de precisión más altos.

XVII-E. Conclusión Técnica

Los resultados obtenidos demuestran que el modelo *Faster R-CNN* es capaz de aprender las características necesarias para la detección de afecciones dentales en imágenes de rayos X, aunque aún requiere mejoras para alcanzar una precisión clínica óptima. Los avances observados son prometedores, y con un conjunto de datos ampliado y mejoras técnicas adicionales, el sistema tiene el potencial de convertirse en una herramienta valiosa en el diagnóstico dental automatizado.

En resumen, aunque el rendimiento actual no es el óptimo, la capacidad de aprendizaje del modelo es clara. Con ajustes y más datos, el *Faster R-CNN* tiene el potencial de convertirse en una solución efectiva para la detección automatizada de afecciones dentales.

XVIII. CRONOGRAMA

A continuación se muestra el cronograma de trabajo en la figura II.

Tabla II
CRONOGRAMA

1. CRONOGRAMA																					
SEDE:	GUAYAQUIL	ESTUDIANTE - AUTOR (1)	FRANCO HERRERA CLAUDIA ANGÉLICA																		
CAMPUS:	CENTENARIO																				
CARRERA:	MECATRÓNICA																				
PERIODO:	2024																				
TEMA DEL TRABAJO DE TITULACIÓN		DESARROLLO DE UN SISTEMA DE ANALISIS DE IMÁGENES DE RAYOS X PARA LA DETECCIÓN DE AFECCIONES DENTALES UTILIZANDO VISIÓN ARTIFICIAL																			
TUTOR DEL TRABAJO DE TITULACIÓN		ING. ALBERTO RAMIREZ																			
Nº	Actividades	Responsable	Estado	Meses/Semanas																	
				Mayo				Junio				Julio				Agosto				Septiembre	
				1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	5	1
	Selección del tema																				
1	Revisión del tema	Claudia Franco	Completado																		
2	Confirmación del tema	Claudia Franco	Completado																		
	Inicio anteproyecto																				
3	Planificación de investigación	Claudia Franco	Completado																		
4	Origen, causa, efecto y problemática	Claudia Franco	Completado																		
5	Base de datos - Artículos	Claudia Franco	Completado																		
6	Planteamiento del problema	Claudia Franco	Completado																		
7	Objetivos - Investigación	Claudia Franco	Completado																		
	Marco teórico referencial																				
8	Justificación del proyecto	Claudia Franco	Completado																		
9	Búsqueda y elaboración del marco teórico	Claudia Franco	Completado																		
10	Marco teórico referencial	Claudia Franco	Completado																		
11	Marco Metodológico	Claudia Franco	Completado																		
12	Propuesta de solución	Claudia Franco	Completado																		
13	Referencias	Claudia Franco	Completado																		
	Trabajo consolidado																				
15	Análisis de presupuesto	Claudia Franco	Completado																		
16	Cronograma de actividades	Claudia Franco	Completado																		
18	Primera revisión consolidado	Claudia Franco	Completado																		
19	Primera corrección consolidado	Claudia Franco	Completado																		
20	Segunda revisión consolidado	Claudia Franco	Completado																		
21	Segunda corrección consolidado	Claudia Franco	Completado																		
22	Compra de equipos a utilizar	Claudia Franco	Completado																		
23	Revisión del sistema de análisis	Claudia Franco	Completado																		
24	Análisis de resultados	Claudia Franco	Completado																		
25	Discusión y conclusiones y anexos	Claudia Franco	Completado																		
26	Presentación del proyecto	Claudia Franco	Completado																		

XIX. PRESUPUESTO

El presupuesto indica los valores aproximados de inversión para la elaboración del proyecto de Tesis.

Desarrollo de un sistema de análisis de imágenes de rayos x para la detección de afecciones dentales utilizando visión artificial.

Costos directos				
Elemento	Tipo de recurso	Unidades	Costo (USD)	Total (USD)
Personal	Sueldo profesionalista	6	400	2400
Equipo de trabajo	Laptop	1	1324	1324
Hojas de papel	Uso continua dentro del proyecto	500	0.008	4
Honorarios profesionales doctores	Servicio	5	80	400
Presupuesto del proyecto				
Presupuesto total invertido			\$4.128,00	

XX. CONCLUSIONES

Se concluye que los resultados obtenidos durante la experimentación evidencian que la tarea de detección automática de afecciones dentales en imágenes de rayos X panorámicas enfrenta desafíos sustanciales, principalmente derivados de la complejidad inherente de estas imágenes. Las radiografías panorámicas contienen una alta densidad de información visual, con estructuras anatómicas superpuestas, variaciones de contraste, y artefactos que complican la identificación precisa de patologías. La naturaleza heterogénea de las características visuales de las afecciones dentales introduce una dificultad considerable en la capacidad del modelo para generalizar y realizar detecciones consistentes.

En particular, las clases correspondientes a diente impactado e implante demostraron ser más fácilmente reconocibles en comparación con caries y lesión periapical. Este hallazgo puede explicarse por la morfología bien definida de los dientes impactados y de los implantes, los cuales presentan contornos más claros y patrones más consistentes que las estructuras difusas asociadas con caries y lesiones periapicales. Las caries y lesiones periapicales, por su naturaleza, tienden a ser más pequeñas y menos definidas, lo que aumenta la complejidad de su detección automática, requiriendo modelos más robustos o métodos adicionales de preprocesamiento.

El uso del modelo Faster R-CNN, basado en redes neuronales convolucionales, demostró ser una alternativa factible para la tarea de detección de afecciones dentales. No obstante, los resultados subóptimos obtenidos sugieren que hay espacio significativo para la mejora. El rendimiento del modelo estuvo limitado por dos factores principales: la cantidad y calidad del dataset y el ajuste de hiperparámetros.

Los hiperparámetros del modelo, incluyendo la tasa de aprendizaje, el número de capas en la red, el tamaño de los batches y los umbrales de clasificación, podrían beneficiarse de un ajuste más fino para mejorar tanto la precisión como la eficiencia del entrenamiento. Aunque el modelo Faster R-CNN es poderoso, la exploración de arquitecturas más especializadas, como las redes profundas diseñadas específicamente para el análisis de imágenes médicas (p. ej., UNet, Mask R-CNN, o DenseNet), podría ofrecer mejoras en la capacidad del sistema para capturar detalles finos en las imágenes.

Finalmente, la implementación de técnicas avanzadas de aumento de datos se presenta como una estrategia clave para mejorar la capacidad del modelo de generalizar sobre datos limitados. El uso de métodos de Data Augmentation que simulen variaciones comunes en las imágenes de rayos X, como rotaciones, escalado, o variaciones de brillo, puede aumentar efectivamente el tamaño del dataset y proporcionar al modelo una mayor diversidad de ejemplos sobre los cuales aprender.

El uso de redes neuronales convolucionales para la detección de afecciones dentales en imágenes de rayos X ha mostrado ser viable, el rendimiento del modelo Faster R-CNN está intrínsecamente ligado a la calidad y cantidad del dataset, así como a los ajustes de los hiperparámetros. Se recomienda la continuación de esta línea de investigación, enfocándose en la optimización del dataset y la exploración de arquitecturas más especializadas, para lograr un sistema robusto y preciso de detección de afecciones dentales.

XXI. RECOMENDACIONES

Es fundamental incrementar el tamaño y balance del dataset utilizado para entrenar el modelo, ya que la cantidad limitada de imágenes y la distribución desigual de las clases dificultaron la capacidad del modelo para generalizar correctamente. Se sugiere recopilar un conjunto de datos más amplio y variado, que incluya más ejemplos de cada clase, especialmente de aquellas que resultan más difíciles de detectar, como las caries y las lesiones periapicales. Asimismo, es crucial asegurar que las imágenes estén adecuadamente etiquetadas y limpias de ruido o artefactos que puedan interferir con el aprendizaje. Incorporar datos de distintas fuentes o con variaciones en las condiciones de adquisición también puede mejorar la capacidad del modelo para reconocer patrones en situaciones reales más diversas.

La optimización de hiperparámetros es otro aspecto clave para mejorar el rendimiento del modelo. Es recomendable realizar una búsqueda exhaustiva de parámetros, ajustando la tasa de aprendizaje, el número de épocas, el tamaño de los lotes (batch size) y el momentum, ya que estos factores influyen directamente en la convergencia del modelo. Utilizar técnicas como el learning rate scheduling o early stopping puede ayudar a alcanzar un equilibrio adecuado entre la velocidad de entrenamiento y la prevención del sobreajuste. Además, explorar diferentes optimizadores, como Adam o RMSprop, podría proporcionar mejoras en el rendimiento del entrenamiento en comparación con el método tradicional de descenso de gradiente estocástico (SGD).

Por último, se recomienda implementar técnicas más avanzadas de aumento de datos para mejorar la robustez del modelo frente a las variaciones naturales en las imágenes de rayos X. Métodos como rotaciones, escalados, ajuste de contraste y brillo, y transformaciones elásticas pueden aumentar la diversidad del dataset sin necesidad de recolectar nuevas imágenes. También se sugiere la exploración de enfoques como el transfer learning, donde un modelo preentrenado en conjuntos de datos más grandes se ajusta específicamente a la tarea de detección dental, lo que podría mejorar significativamente los resultados. Además, considerar la utilización de arquitecturas especializadas en el análisis de imágenes médicas, como UNet o DenseNet, podría proporcionar ventajas en la detección precisa de afecciones dentales.

REFERENCIAS

- [1] G. Parise Zambrano, «Estado de salud bucal en Ecuador,» 2017.
- [2] Condori, *Sistema experto para el diagnostico de enfermedades bucales prevenibles para el consultorio dental Virgen de remedios*, 2016.
- [3] D. S. Quito Cárdenas y V. L. Salvatierra Guanuche, *DIAGNÓSTICO PRELIMINAR MEDIANTE PROCESAMIENTO DIGITAL DE IMÁGENES PARA PATOLOGÍAS PRESENTES EN LA CAVIDAD BUCAL*, 2018.
- [4] T. C. C. D. L. Chonata, *Situación actual de la salud oral en el Ecuador*, 2017. dirección: <https://sesp-ec.com/situacion-actual-de-la-salud-oral-en-el-ecuador/>.
- [5] B. E., *El impacto de la Inteligencia Artificial en la Odontología*, 2023. dirección: <https://la.dental-tribune.com/news/el-impacto-de-la-inteligencia-artificial-en-la-odontologia/>.
- [6] C. G. G. D. L. Victor, «Aplicación de la inteligencia artificial en odontología,» pág. 8, 2021.
- [7] H. R. Natalia, *Comparación de la precisión de escaneado en rehabilitaciones sobre implantes mediante escáner intraoral vs escáner intraoral combinado con fotogrametría: una revisión sistemática*, 2023.
- [8] OMS, «La OMS destaca que el descuido de la salud bucodental afecta a casi la mitad de la población mundial,» 2022.
- [9] E. J. Bechara, *Redes neuronales convolucionales: un modelo de Deep Learning en imágenes diagnósticas. Revisión de tema*, 2023. dirección: <https://la.dental-tribune.com/news/el-impacto-de-la-inteligencia-artificial-en-la-odontologia/>.
- [10] Álvaro Artola Moreno, *Clasificación de imágenes usando redes neuronales convolucionales en Python*, 2019.
- [11] Xeridia, *Redes Neuronales artificiales: Qué son y cómo se entrenan*. 2023. dirección: <https://www.xeridia.com/blog/redes-neuronales-artificiales-que-son-y-como-se-entrenan-parte-i>.
- [12] D. M. C. B. Dra. Marianella Álvarez Vega Dra. Laura Quirós Mora, «Inteligencia artificial y aprendizaje automático en medicina,» pág. 4, 2020.
- [13] I. E. Hamamci, S. Er, E. Simsar et al., *Diffusion-Based Hierarchical Multi-Label Object Detection to Analyze Panoramic Dental X-rays*, 2023.
- [14] S. S. L. Judith, «Algoritmos de aprendizaje automático para análisis y predicción de datos,» págs. 2-3, 2018.
- [15] C. Diego, *Clasificación de redes neuronales artificiales*, 2019. dirección: <https://www.xeridia.com/blog/redes-neuronales-artificiales-que-son-y-como-se-entrenan-parte-i>.
- [16] G. M. Aurea Castro, *La inteligencia artificial en el análisis de imágenes radiológicas y el diagnóstico de enfermedades En el Hospital Juan Montalván del cantón Urdaneta perteneciente a la parroquia Ricaurte*. 2022.
- [17] Pearl, *DT News - Latin America - El primer software de AI del ...* Mayo de 2022. dirección: <https://la.dental-tribune.com/news/el-primer-software-de-ai-del-mundo-que-interpreta-radiografias-dentales/>.
- [18] J. Miller y S. Roberts, «Advancements in Dental Radiography,» *Journal of Dental Research*, vol. 99, n.º 5, págs. 123-130, 2020.
- [19] A. Smith, *Comprehensive Guide to Panoramic Radiography*, 2nd. New York: Medical Imaging Press, 2021.
- [20] J. Y. Cha, A. S. B. B. K. R. B. R. A. B. C. A. e Y. D. Lee, «Artificial intelligence in the diagnosis of dental caries: A systematic review,» *Journal of Dental Research*, vol. 100, n.º 4, págs. 464-472, 2021. DOI: 10.1177/00220345221027219.
- [21] M. Peric, L. G. M. M. K. L. P. y M. D. P. R. P., «Lesion periapical management in endodontics: Current challenges and new frontiers,» *International Endodontic Journal*, vol. 54, n.º 6, págs. 1043-1053, 2021. DOI: 10.1111/iej.13456.
- [22] G. Mendes, R. M. G. E. S. F. R. y F. S. R., «Impacted teeth: Diagnosis and surgical management using cone-beam CT,» *Oral Surgery, Oral Medicine, Oral Pathology, Oral Radiology*, vol. 130, n.º 2, págs. 141-148, 2020. DOI: 10.1016/j.oooo.2020.02.014.
- [23] J. H. Ryu, J. H. M. J. K. H. y S. K. Lee, «Dental implant success: Advances in surgical technology and biomaterials,» *Journal of Clinical Periodontology*, vol. 47, n.º 10, págs. 1053-1062, 2020. DOI: 10.1111/jcpe.13226.
- [24] Y. LeCun, Y. Bengio y G. Hinton, «Deep Learning,» *Nature*, vol. 521, n.º 7553, págs. 436-444, 2015.

- [25] S. Ren, K. He, R. B. Girshick y J. Sun, «Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks,» *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, n.º 6, págs. 1137-1149, 2015.
- [26] J. Redmon, S. Divvala, R. Girshick y A. Farhadi, «You Only Look Once: Unified, Real-Time Object Detection,» *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, págs. 779-788, 2016.
- [27] I. Goodfellow, J. Pouget-Abadie, M. Mirza et al., «Generative Adversarial Nets,» *Advances in Neural Information Processing Systems*, vol. 27, 2014.
- [28] M. Fridman y R. Hartley, «Synthetic Data Generation for Augmentation of Medical Imaging Datasets,» *Medical Image Analysis*, vol. 48, págs. 37-46, 2018.
- [29] A. Radford, L. Metz y S. Chintala, «Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks,» *Proceedings of the International Conference on Learning Representations*, 2015.
- [30] T. Karras, S. Laine y T. Aila, «A Style-Based Generator Architecture for Generative Adversarial Networks,» *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, págs. 4401-4410, 2019.
- [31] Y. Kim y S.-Y. Lee, «Variability in Dental Radiographic Quality and Its Implications,» *Journal of Medical Imaging*, vol. 29, n.º 2, págs. 234-245, 2022.
- [32] J. Lee y M. Johnson, «Preprocessing Techniques in Medical Image Analysis,» *Biomedical Engineering Reviews*, vol. 45, n.º 3, págs. 289-300, 2023.
- [33] L. Wang, X. Zhang y M. Chen, «Automated Annotation Tools for Medical Image Datasets,» *Journal of Healthcare Informatics*, vol. 56, n.º 1, págs. 105-116, 2021.
- [34] L. Contributors, *Labelme: A Graphical Image Annotation Tool*, Labelme, 2018. dirección: <https://github.com/wkentaro/labelme>.
- [35] A. Inc., *Anaconda Distribution*, 2024. dirección: <https://www.anaconda.com/products/distribution>.
- [36] C. Team, *Conda Documentation*, 2024. dirección: <https://docs.conda.io/projects/conda/en/latest/>.

ANEXO A PROYECTO

Código de entrenamiento completo:

```
[1]: import torch
print(f"PyTorch version: {torch.__version__}")
print(f"CUDA version in PyTorch: {torch.version.cuda}")
print("CUDA is available:", torch.cuda.is_available())

PyTorch version: 2.3.1
CUDA version in PyTorch: 12.1
CUDA is available: True
```

Figura 46. Verificación de GPU habilitada por C. Franco, Jupyter Notebook.

```
[2]: !pip install opencv-python opencv-python-headless
!pip install matplotlib
!pip install scikit-learn
!pip install scikit-image
!pip install albumentations
```

Figura 47. Instalación de librerías necesarias por C. Franco, Jupyter Notebook.

```
[3]: import torch
import torchvision
import cv2
import matplotlib
import skimage
import albumentations

#COMPROBAR EMPLENADO UN ARCHIVO PYTHON
print(f"PyTorch version: {torch.__version__}")
print(f"TorchVision version: {torchvision.__version__}")
print(f"OpenCV version: {cv2.__version__}")
print(f"Matplotlib version: {matplotlib.__version__}")
print(f"scikit-image version: {skimage.__version__}")
print(f"Albumentations version: {albumentations.__version__}")
# Verificar si la GPU está disponible
gpu = torch.cuda.is_available()
print("GPU is", "available" if gpu else "NOT AVAILABLE")

PyTorch version: 2.3.1
TorchVision version: 0.18.1
OpenCV version: 4.10.0
Matplotlib version: 3.7.2
scikit-image version: 0.21.0
Albumentations version: 1.4.14
GPU is available
```

Figura 48. Comprobación de GPU habilitada por C. Franco, Jupyter Notebook.

```
[4]: pip install --upgrade pip setuptools wheel

Requirement already satisfied: pip in c:\users\claudia franco\anaconda3\envs\pytorchb_env\lib\site-packages (24.2)
Requirement already satisfied: setuptools in c:\users\claudia franco\anaconda3\envs\pytorchb_env\lib\site-packages (72.1)
Collecting setuptools
  Using cached setuptools-74.1.2-py3-none-any.whl.metadata (6.7 kB)
Requirement already satisfied: wheel in c:\users\claudia franco\anaconda3\envs\pytorchb_env\lib\site-packages (0.43.0)
Collecting wheel
  Using cached wheel-0.44.0-py3-none-any.whl.metadata (2.3 kB)
Using cached setuptools-74.1.2-py3-none-any.whl (1.3 MB)
Using cached wheel-0.44.0-py3-none-any.whl (67 kB)
Installing collected packages: wheel, setuptools
  Attempting uninstall: wheel
    Found existing installation: wheel 0.43.0
    Uninstalling wheel-0.43.0:
      Successfully uninstalled wheel-0.43.0
  Attempting uninstall: setuptools
    Found existing installation: setuptools 72.1.0
    Uninstalling setuptools-72.1.0:
      Successfully uninstalled setuptools-72.1.0
Successfully installed setuptools-74.1.2 wheel-0.44.0
Note: you may need to restart the kernel to use updated packages.
```

Figura 49. Instalación de herramientas por C. Franco, Jupyter Notebook.

```
[5]: !pip install torchmetrics
```

```
Requirement already satisfied: torchmetrics in c:\users\claudia franco\anaconda3\envs\pytorchb_env\lib\site-packages
Requirement already satisfied: numpy>1.20.0 in c:\users\claudia franco\anaconda3\envs\pytorchb_env\lib\site-packages
Requirement already satisfied: packaging>17.1 in c:\users\claudia franco\anaconda3\envs\pytorchb_env\lib\site-packages
Requirement already satisfied: torch>=1.10.0 in c:\users\claudia franco\anaconda3\envs\pytorchb_env\lib\site-packages
Requirement already satisfied: lightning-utilities>=0.8.0 in c:\users\claudia franco\anaconda3\envs\pytorchb_env\lib\
(0.11.6)
Requirement already satisfied: typing-extensions in c:\users\claudia franco\anaconda3\envs\pytorchb_env\lib\site-pack
Requirement already satisfied: setuptools in c:\users\claudia franco\anaconda3\envs\pytorchb_env\lib\site-packages (f
hmetrics) (74.1.2)
Requirement already satisfied: filelock in c:\users\claudia franco\anaconda3\envs\pytorchb_env\lib\site-packages (fr
3.1)
Requirement already satisfied: sympy in c:\users\claudia franco\anaconda3\envs\pytorchb_env\lib\site-packages (from t
Requirement already satisfied: networkx in c:\users\claudia franco\anaconda3\envs\pytorchb_env\lib\site-packages (fr
Requirement already satisfied: Jinja2 in c:\users\claudia franco\anaconda3\envs\pytorchb_env\lib\site-packages (from
Requirement already satisfied: fsspec in c:\users\claudia franco\anaconda3\envs\pytorchb_env\lib\site-packages (from
6.1)
Collecting mkl<=2021.4.0,>=2021.1.1 (from torch>=1.10.0->torchmetrics)
  Using cached mkl-2021.4.0-py2.py3-none-win_amd64.whl.metadata (1.4 kB)
Collecting intel-openmp==2021.* (from mkl<=2021.4.0,>=2021.1.1->torch>=1.10.0->torchmetrics)
  Using cached intel_openmp-2021.4.0-py2.py3-none-win_amd64.whl.metadata (1.2 kB)
Collecting tbb==2021.* (from mkl<=2021.4.0,>=2021.1.1->torch>=1.10.0->torchmetrics)
  Using cached tbb-2021.13.1-py3-none-win_amd64.whl.metadata (1.1 kB)
Requirement already satisfied: MarkupSafe>=2.0 in c:\users\claudia franco\anaconda3\envs\pytorchb_env\lib\site-packag
hmetrics) (2.1.3)
Requirement already satisfied: mpmath>=0.19 in c:\users\claudia franco\anaconda3\envs\pytorchb_env\lib\site-packages
rics) (1.3.0)
Using cached mkl-2021.4.0-py2.py3-none-win_amd64.whl (228.5 MB)
Using cached intel_openmp-2021.4.0-py2.py3-none-win_amd64.whl (3.5 MB)
```

Figura 50. Instalación de métricas por C. Franco, Jupyter Notebook.

```
[1]: import json
import os
from PIL import Image
import torch
from torch.utils.data import random_split
from torchvision import transforms
from tqdm import tqdm
import time
from datetime import datetime
import torchmetrics
from torchmetrics.detection.mean_ap import MeanAveragePrecision
import matplotlib.pyplot as plt

metric = torchmetrics.detection.MeanAveragePrecision(class_metrics=True)

def validate_one_epoch(model, loader, device):
    model.eval()
    metric.reset()

    with torch.no_grad():
        for images, targets in tqdm(loader):
            images = list(image.to(device) for image in images)
```

Figura 51. Importación de librerías para entrenamiento por C. Franco, Jupyter Notebook.

```
metric = torchmetrics.detection.MeanAveragePrecision(class_metrics=True)

def validate_one_epoch(model, loader, device):
    model.eval()
    metric.reset()

    with torch.no_grad():
        for images, targets in tqdm(loader):
            images = list(image.to(device) for image in images)
            targets = [{k: v.to(device) for k, v in t.items()} for t in targets]
            outputs = model(images)

            # Asegurarse de que los outputs estén en el dispositivo correcto
            for output in outputs:
                output['boxes'] = output['boxes'].to(device)
                output['scores'] = output['scores'].to(device)
                output['labels'] = output['labels'].to(device)

            # Mover los targets también al dispositivo correcto
            targets = [{k: v.to(device) for k, v in t.items()} for t in targets]

            metric.update(outputs, targets)

results = metric.compute()
```

Figura 52. código de métricas 1 por C. Franco, Jupyter Notebook.

```

# Verificar map_per_class
if 'map_per_class' in results:
    map_per_class = results['map_per_class']
    if map_per_class.ndim > 0: # Si es un tensor de más de 0 dimensiones
        for i, ap in enumerate(map_per_class):
            print(f"Class {i + 1}: mAP = {ap:.4f}")
    else:
        print(f"mAP for the only class: {map_per_class.item():.4f}")
else:
    # Métricas
    if 'map_50_per_class' in results:
        map_50_per_class = results['map_50_per_class']
        if map_50_per_class.ndim > 0: # Si es un tensor de más de 0 dimensiones
            for i, ap in enumerate(map_50_per_class):
                print(f"Class {i + 1}: mAP@50 = {ap:.4f}")
        else:
            print(f"mAP@50 for the only class: {map_50_per_class.item():.4f}")
    else:
        print("No se encontraron métricas por clase en los resultados.")

return results

```

Figura 53. código de métricas 2 por C. Franco, Jupyter Notebook.

```

def calculate_accuracy(predictions, targets):
    correct = sum(p == t for p, t in zip(predictions, targets))
    accuracy = correct / len(targets)
    return accuracy

def load_labelme_annotation(json_path, class_mapping):
    with open(json_path) as f:
        data = json.load(f)

    boxes, labels = [], []

    for shape in data['shapes']:
        points = shape['points']
        xmin = min(points[0][0], points[1][0])
        ymin = min(points[0][1], points[1][1])
        xmax = max(points[0][0], points[1][0])
        ymax = max(points[0][1], points[1][1])

        label_name = shape['label']
        if label_name in class_mapping:
            label = class_mapping[label_name]
        else:
            continue # Ignora si no está en el mapeo

        if xmax > xmin and ymax > ymin:
            boxes.append([xmin, ymin, xmax, ymax])
            labels.append(label)

```

Figura 54. Código dataset 1 por C. Franco, Jupyter Notebook.

```

# Verifica que Las etiquetas estén dentro del rango correcto
for label in labels:
    assert 0 <= label < len(class_mapping), f"Etiqueta {label} fuera de rango."

return boxes, labels

class LabelmeDataset(torch.utils.data.Dataset):
    def __init__(self, img_dir, annotation_dir, class_mapping, transforms=None):
        self.img_dir = img_dir
        self.annotation_dir = annotation_dir
        self.transforms = transforms
        self.class_mapping = class_mapping # Guardar el class_mapping como parte del dataset
        self.imgs = list(sorted(os.listdir(img_dir)))
        self.anns = list(sorted(os.listdir(annotation_dir)))
        self.empty_boxes_imgs = []

    def __getitem__(self, idx):
        img_path = os.path.join(self.img_dir, self.imgs[idx])
        ann_path = os.path.join(self.annotation_dir, self.anns[idx])

        img = Image.open(img_path).convert("RGB")
        boxes, labels = load_labelme_annotation(ann_path, self.class_mapping) # Pasar class_mapping aquí

        if len(boxes) == 0:
            self.empty_boxes_imgs.append((img_path, ann_path))
            return None, None

        boxes = torch.as_tensor(boxes, dtype=torch.float32)
        labels = torch.as_tensor(labels, dtype=torch.int64)

```

Figura 55. Código dataset 2 por C. Franco, Jupyter Notebook.

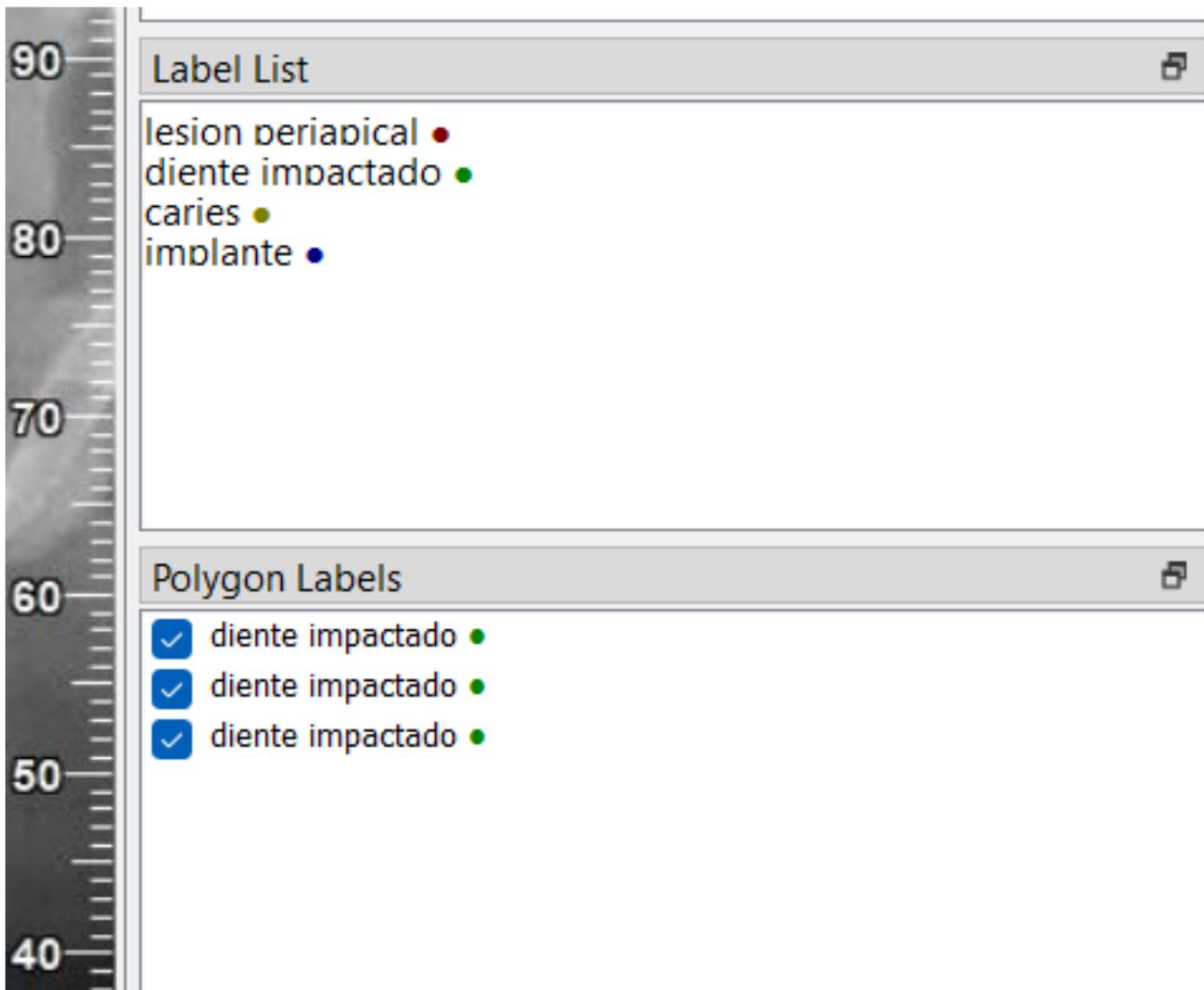


Figura 56. Clasificación de etiquetas 1 por C. Franco, Labelme.

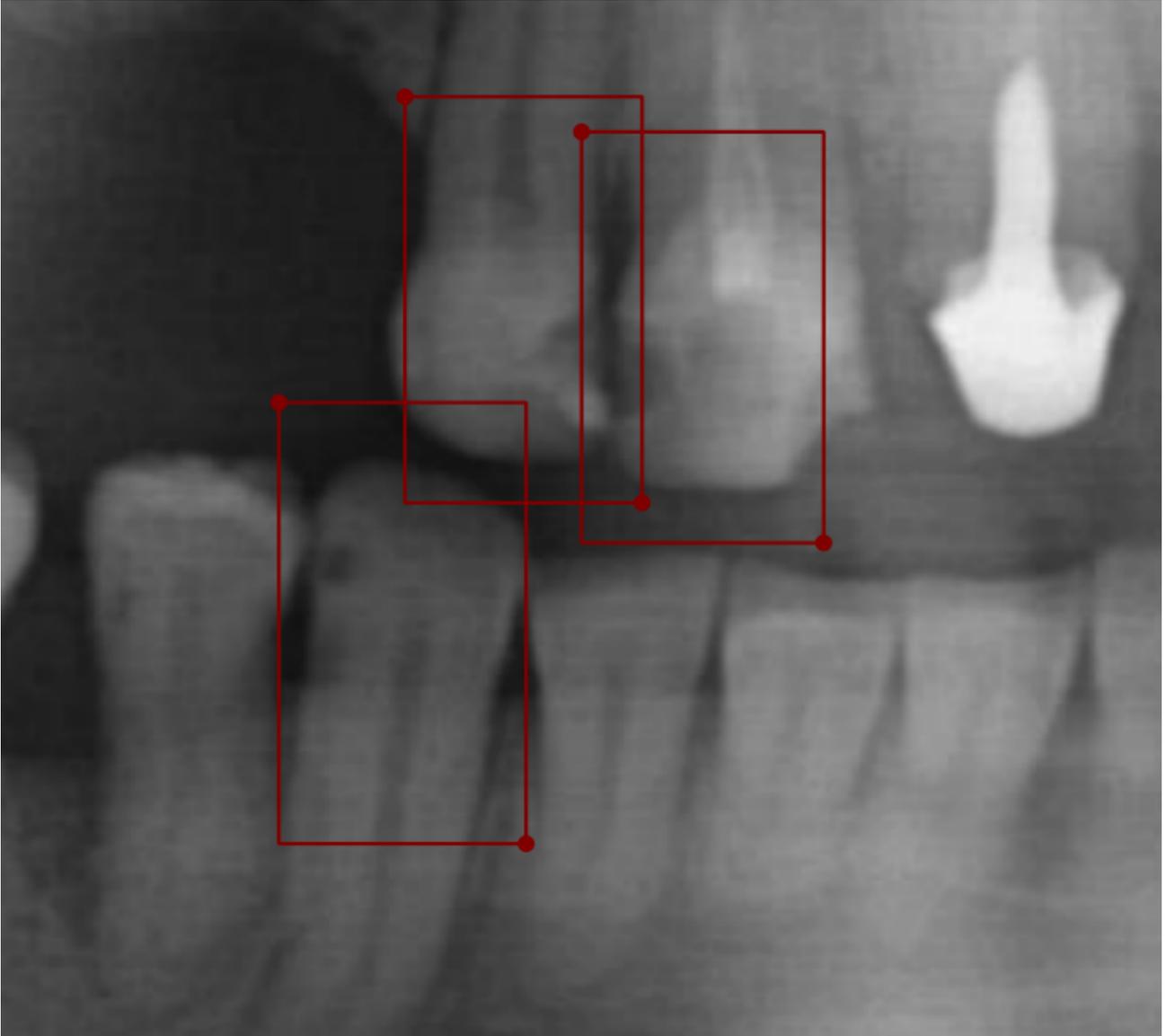


Figura 57. Etiquetado incorrecto demostración de etiqueta rectangular por C. Franco, Labelme.

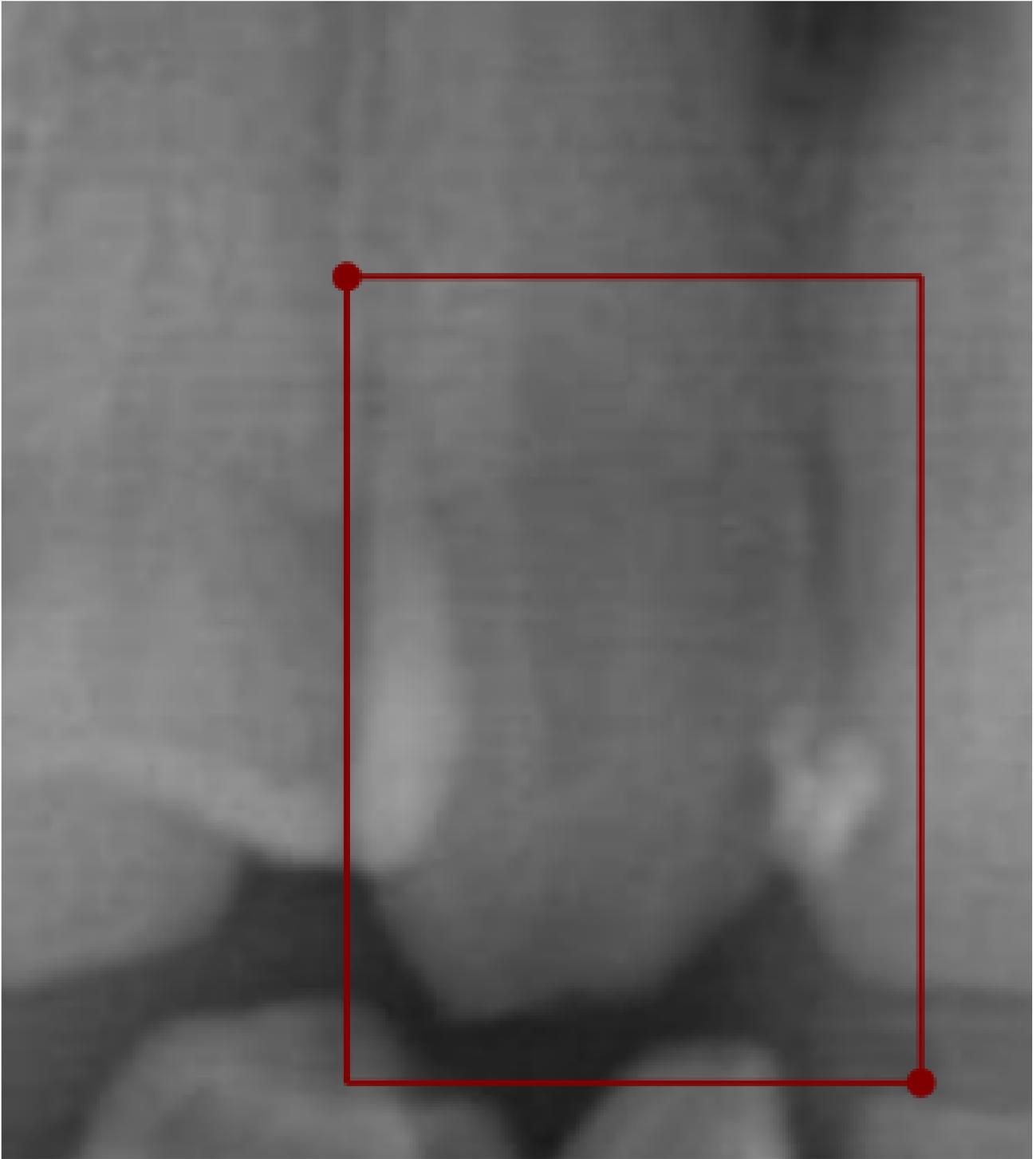


Figura 58. Etiquetado correcto demostración de etiqueta rectangular por C. Franco, Labelme

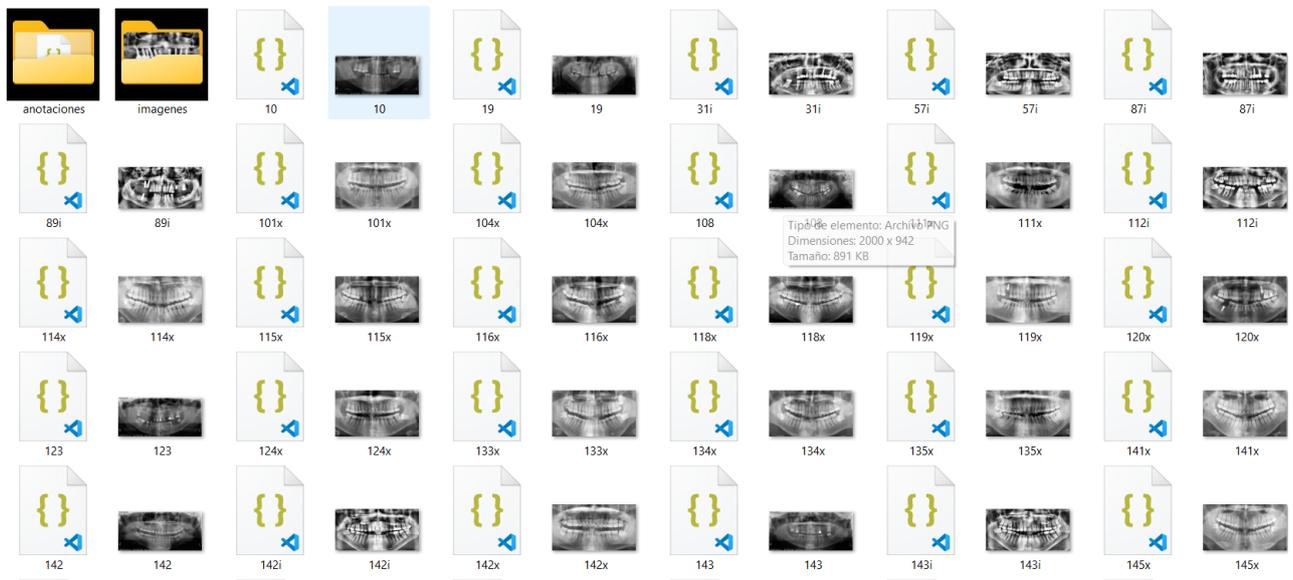


Figura 59. Imágenes panorámicas con etiquetas por C. Franco, dataset

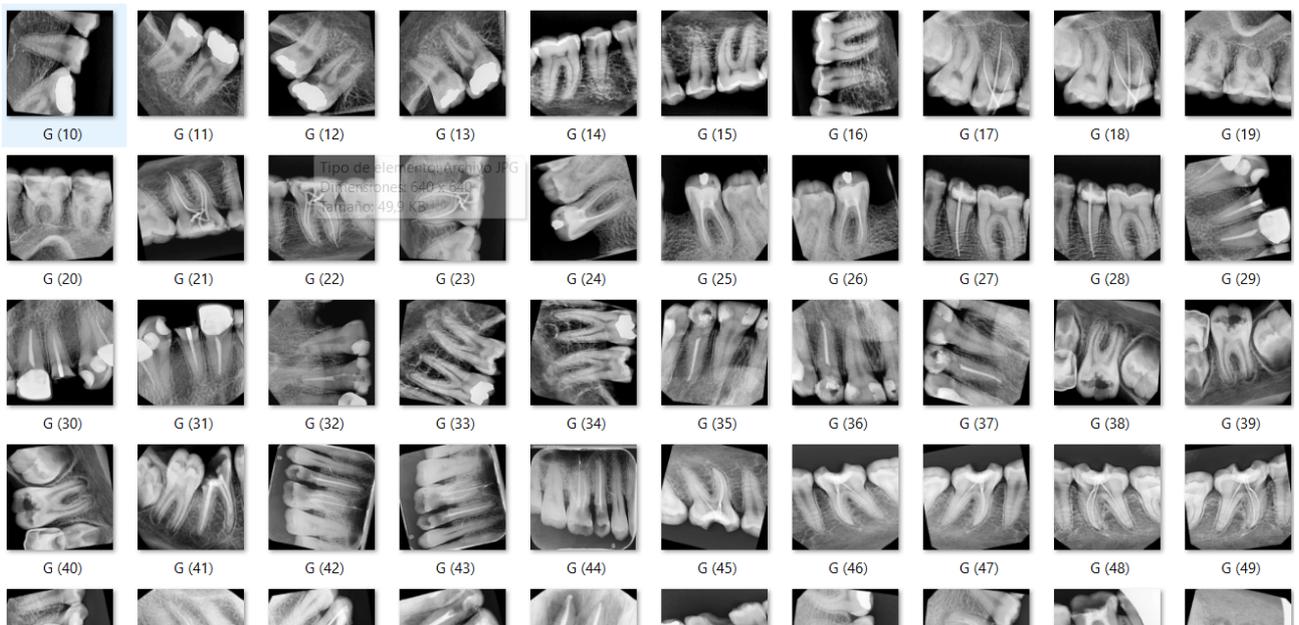


Figura 60. Imágenes periapical por C. Franco, dataset

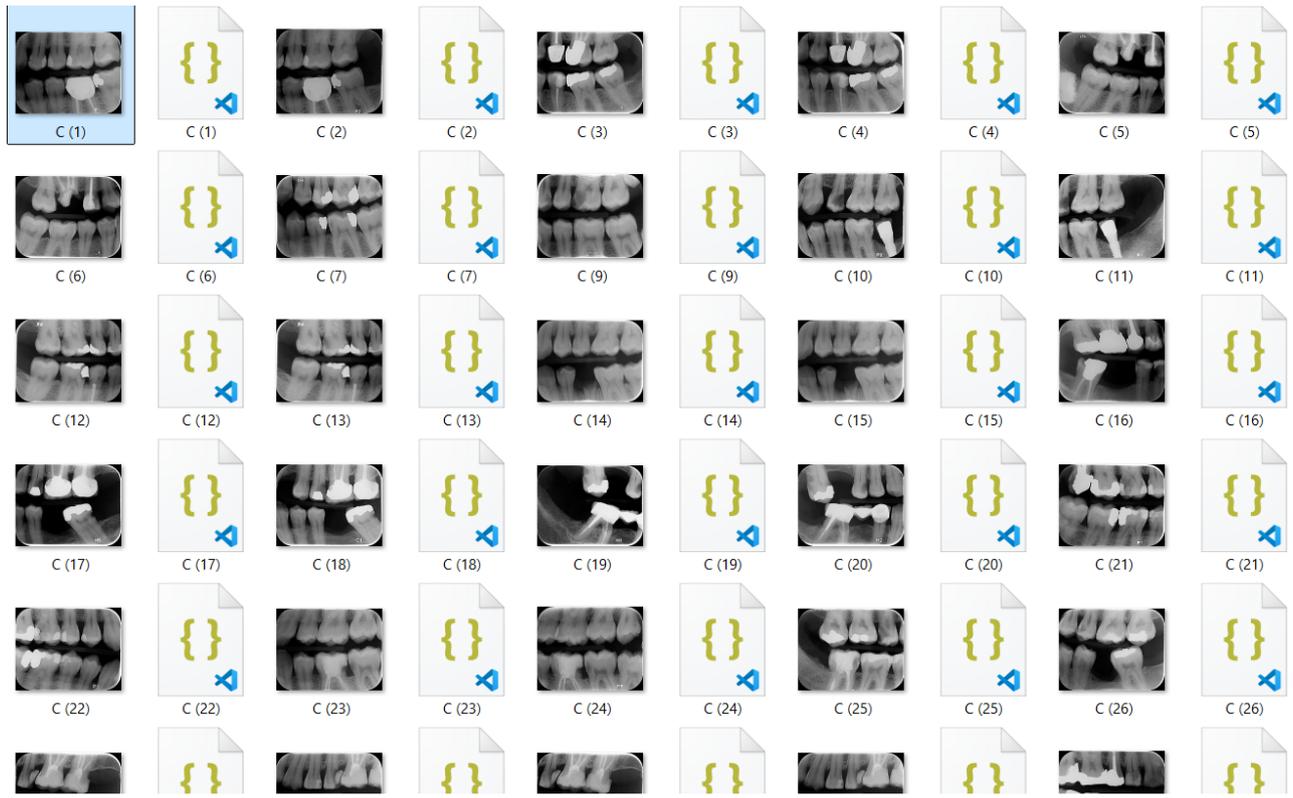


Figura 61. Imágenes periapical 2 con etiquetas por C. Franco, dataset.

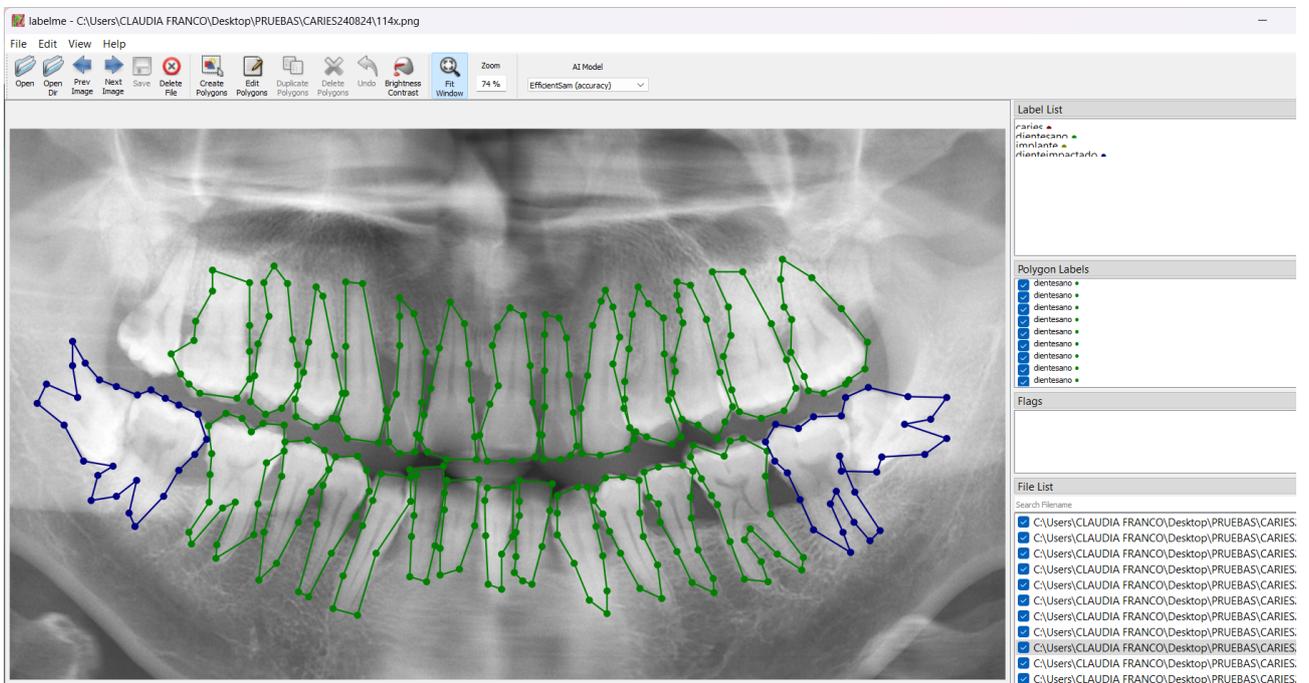


Figura 62. Etiquetado poligonal para todo tipo de dientes, por C. Franco, Labelme

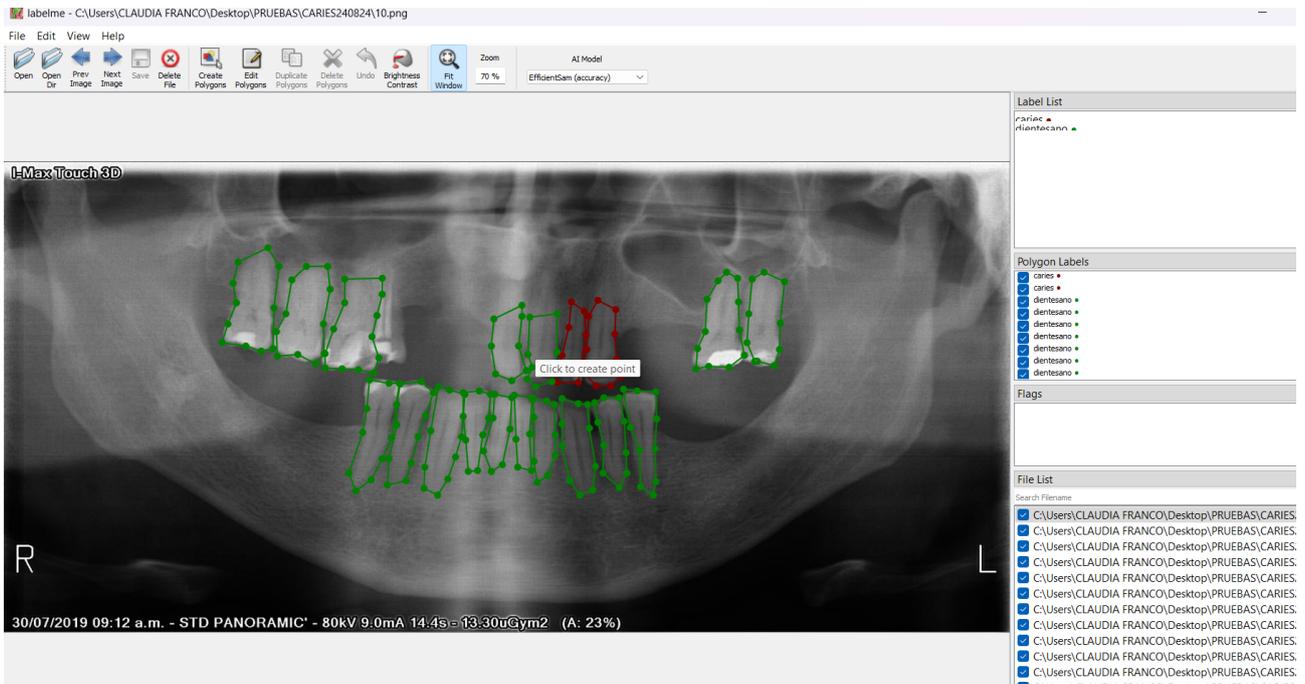


Figura 63. Etiquetado poligonal 2 para todo tipo de dientes, por C. Franco, Labelme

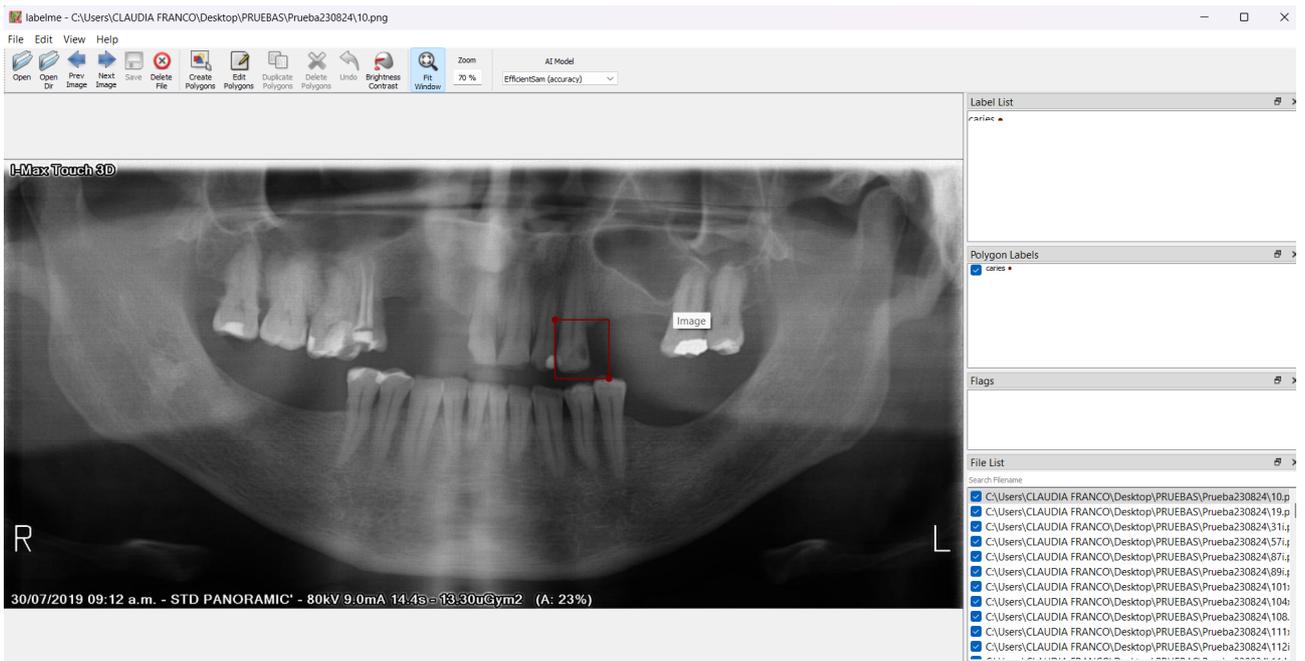


Figura 64. Etiquetado rectangular imagen panorámica, por C. Franco, Labelme.

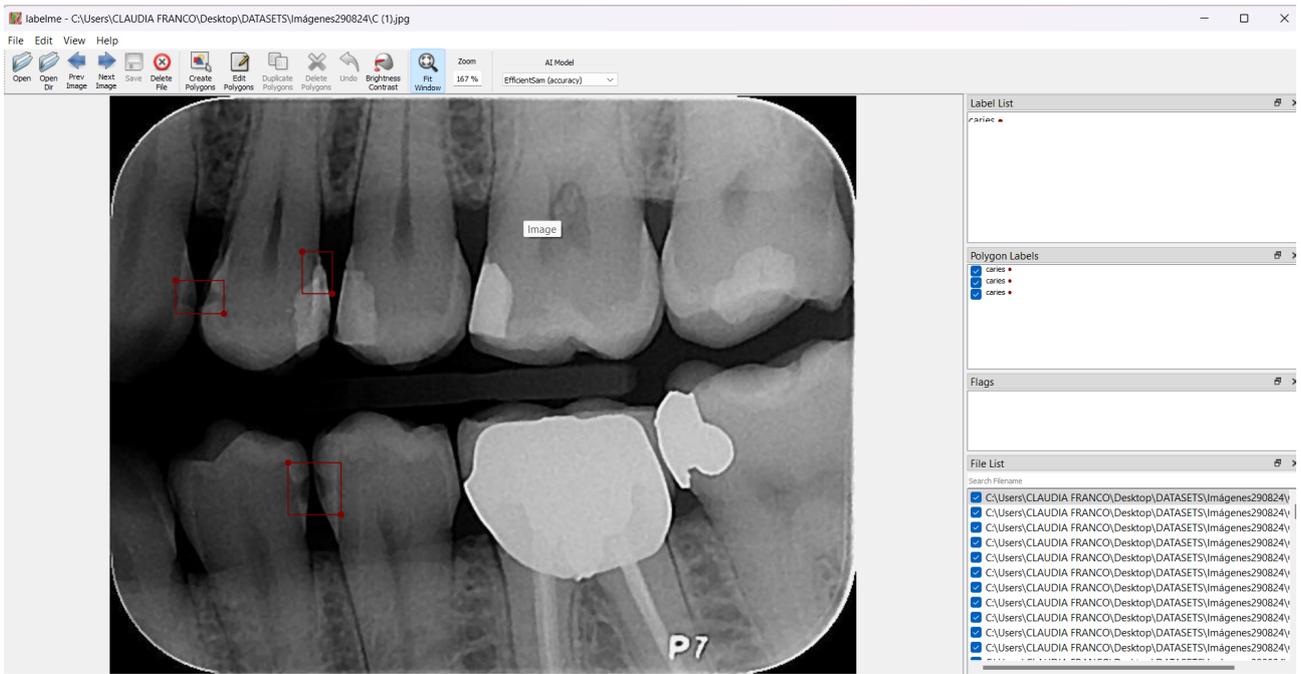


Figura 65. Etiquetado rectangular imagen periapical, por C. Franco, Labelme.

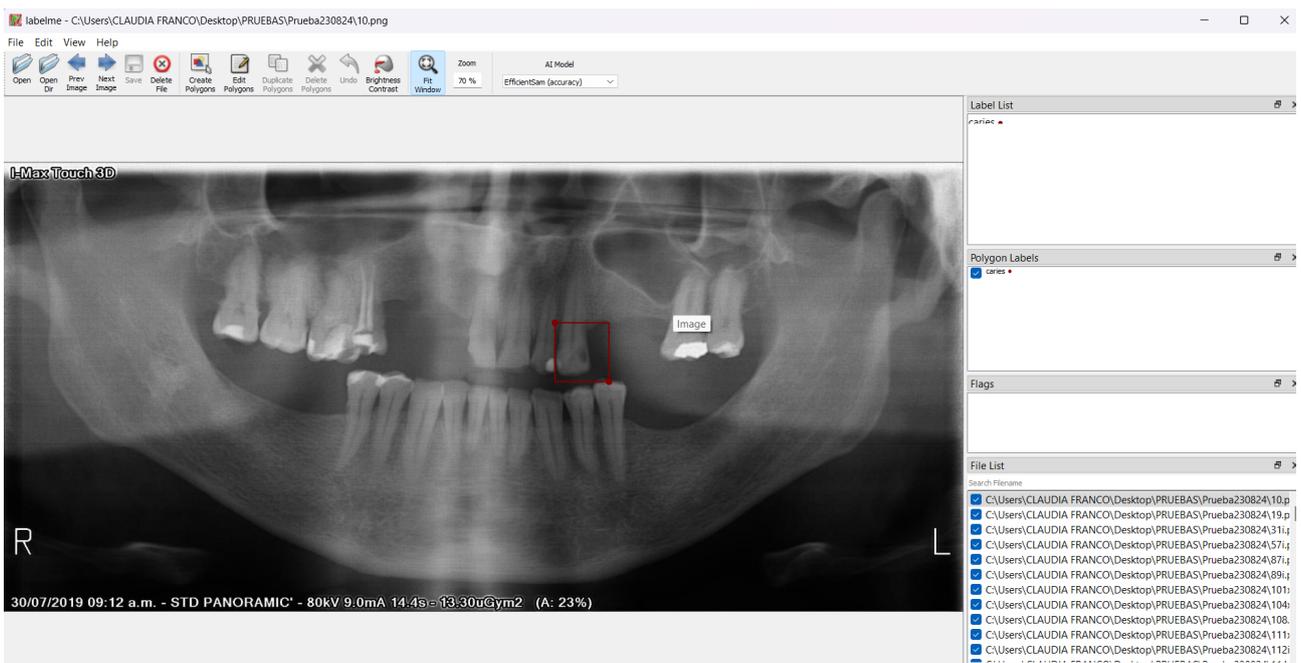


Figura 66. Etiquetado rectangular imagen panoramica, por C. Franco, Labelme.

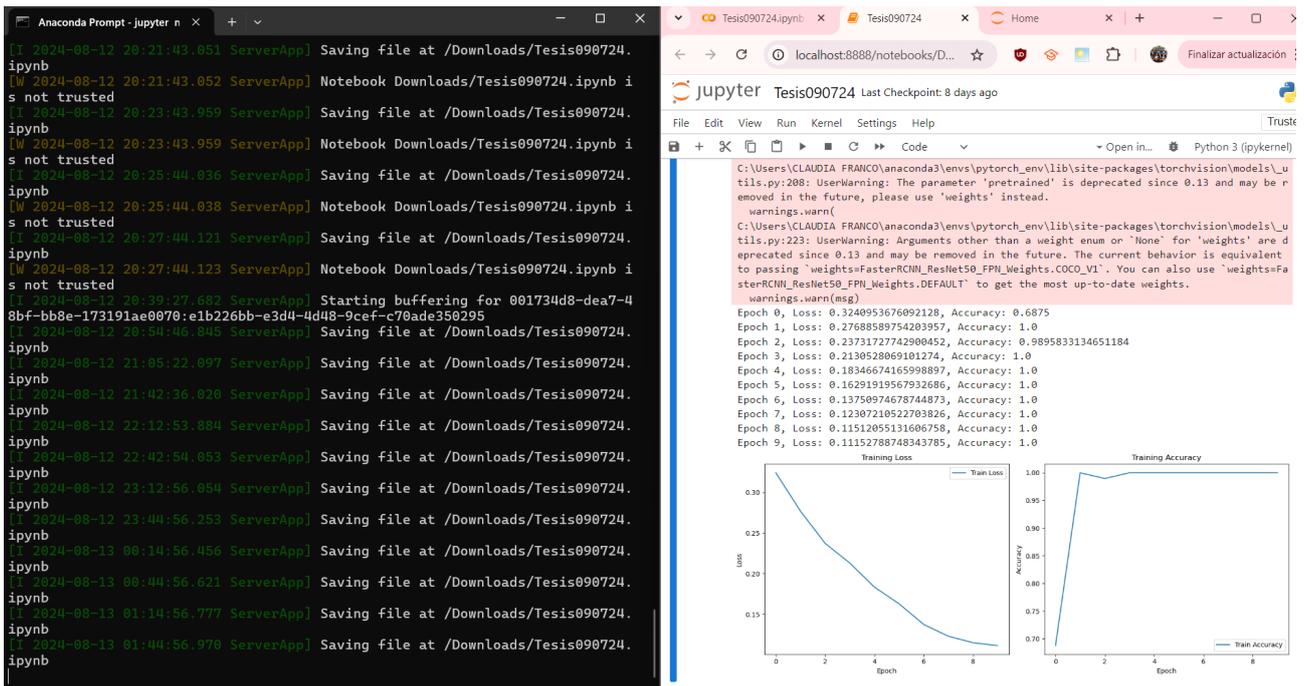


Figura 67. Pruebas realizadas, por C. Franco, conda.



Figura 68. Etiquetado de imágenes con el profesional