

# UNIVERSIDAD TÉCNICA DEL NORTE



Facultad de Ingeniería en Ciencias Aplicadas  
Carrera de Ingeniería en Sistemas Computacionales

## **ANÁLISIS DE IMÁGENES DE RADIOGRAFÍAS DE LA COLUMNA VERTEBRAL PARA CALCULAR EL ÁNGULO DE DESVIACIÓN EN PACIENTES QUE PADECEN ESCOLIOSIS UTILIZANDO TÉCNICAS DE IA.**

Trabajo de grado previo a la obtención del título de Ingeniero en Sistemas  
Computacionales

Autor:

Carlos Antonio Cumbal Sánchez

Director:

PhD. Iván Danilo García Santillán

Ibarra - Ecuador

2022

# AUTORIZACIÓN DE USO Y PUBLICACIÓN A FAVOR DE LA UNIVERSIDAD



## UNIVERSIDAD TÉCNICA DEL NORTE BIBLIOTECA UNIVERSITARIA

### AUTORIZACIÓN DE USO Y PUBLICACIÓN A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE

#### 1. IDENTIFICACIÓN DE LA OBRA

En cumplimiento del Art. 144 de la Ley de Educación Superior, hago la entrega del presente trabajo a la Universidad Técnica del Norte para que sea publicado en el Repositorio Digital Institucional, para lo cual pongo a disposición la siguiente información:

DATOS DE CONTACTO			
CÉDULA DE IDENTIDAD:	1755224423		
APELLIDOS Y NOMBRES:	Cumbal Sánchez Carlos Antonio		
DIRECCIÓN:	Ayora – Cayambe - Pichincha		
EMAIL:	cacumbals@utn.edu.ec, carloscsa1998@gmail.com		
TELÉFONO FIJO:	-	TELÉFONO MÓVIL:	0963480029

DATOS DE LA OBRA	
TÍTULO:	ANÁLISIS DE IMÁGENES DE RADIOGRAFÍAS DE LA COLUMNA VERTEBRAL PARA CALCULAR EL ÁNGULO DE DESVIACIÓN EN PACIENTES QUE PADECEN ESCOLIOSIS UTILIZANDO TÉCNICAS DE IA.
AUTOR (ES):	Cumbal Sánchez Carlos Antonio
FECHA: DD/MM/AAAA	25/10/2022
SOLO PARA TRABAJOS DE GRADO	
PROGRAMA:	<input checked="" type="checkbox"/> PREGRADO <input type="checkbox"/> POSGRADO
TÍTULO POR EL QUE OPTA:	INGENIERIA EN SISTEMAS COMPUTACIONALES
ASESOR /DIRECTOR:	PhD. Iván Danilo García Santillán

#### 2. CONSTANCIAS

El autor (es) manifiesta (n) que la obra objeto de la presente autorización es original y se la desarrolló, sin violar derechos de autor de terceros, por lo tanto la obra es original y que es (son) el (los) titular (es) de los derechos patrimoniales, por lo que asume (n) la responsabilidad sobre el contenido de la misma y saldrá (n) en defensa de la Universidad en caso de reclamación por parte de terceros.

Ibarra, a los 25 días del mes de octubre de 2022

EL AUTOR:

(Firma).....

Nombre: Carlos Cumbal

## CERTIFICACIÓN DEL DIRECTOR DE TRABAJO DE GRADO



UNIVERSIDAD TÉCNICA DEL NORTE  
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS  
CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES

### CERTIFICADO DEL DIRECTOR

En mi calidad de Tutor de Trabajo de Grado presentado por el egresado, **Carlos Antonio Cumbal Sánchez** para optar por el Título de Ingeniero en Sistemas Computacionales, cuyo tema es: **ANÁLISIS DE IMÁGENES DE RADIOGRAFÍAS DE LA COLUMNA VERTEBRAL PARA CALCULAR EL ÁNGULO DE DESVIACIÓN EN PACIENTES QUE PADECEN ESCOLIOSIS UTILIZANDO TÉCNICAS DE IA**. Considero que el presente trabajo reúne los requisitos y méritos suficientes para ser sometido a la presentación pública y evaluación por parte del tribunal examinador.

En la ciudad de Ibarra, a los 21 días del mes de octubre del 2022.

A handwritten signature in blue ink, appearing to read "Iván Danilo García Santillán", is written over a horizontal line.

PhD. Iván Danilo García Santillán  
**DIRECTOR DE TRABAJO DE GRADO**

## CERTIFICACIÓN INSTITUCIÓN BENEFICIARIA



### CERTIFICADO

QUE: El Sr. CARLOS ANTONIO CUMBAL SÁNCHEZ con cédula de identidad 1755224423 estudiante de la Universidad Técnica del Norte de la Facultad de Ingeniería en Ciencias Aplicadas de la Carrera de Ingeniería en Sistemas Computacionales, ha desarrollado con los datos entregados por el Centro de Rayos X el Proyecto de Tesis: **“ANÁLISIS DE IMÁGENES DE RADIOGRAFÍAS DE LA COLUMNA VERTEBRAL PARA CALCULAR EL ÁNGULO DE DESVIACIÓN EN PACIENTES QUE PADECEN ESCOLIOSIS UTILIZANDO TÉCNICAS DE IA.”**

QUE: El proyecto desarrollado fue entregado y revisado por el Centro de Rayos X en el Cantón Cayambe el 14 de agosto del 2022.

Es todo cuanto puedo certificar, facultando al Interesado hacer uso de este certificado como estime conveniente, excepto para trámites judiciales.

Cayambe, 14 de agosto del 2022

Atentamente



Lic. Wilson Cataña  
Dueño del Centro de Rayos X

## **DEDICATORIA**

Por el intenso apoyo, comprensión, amor, esfuerzo y fe en lo que en un futuro seré, este proyecto de titulación va dedicado a mi madre.

## **AGRADECIMIENTO**

A mi familia, por el inmenso apoyo, por sus palabras de motivación durante mi carrera universitaria. A mis compañeros por los buenos momentos y el gran equipo que formamos para apoyarnos mutuamente, agradezco a los docentes por su conocimiento brindado.

## Tabla de Contenido

AUTORIZACIÓN DE USO Y PUBLICACIÓN A FAVOR DE LA UNIVERSIDAD.....	II
CERTIFICACIÓN DEL DIRECTOR DE TRABAJO DE GRADO .....	III
CERTIFICACIÓN INSTITUCIÓN BENEFICIARIA .....	IV
DEDICATORIA.....	V
AGRADECIMIENTO .....	VI
Introducción.....	XIV
Problema.....	XIV
Objetivos .....	XVI
Alcance .....	XVI
Justificación .....	XVIII
CAPÍTULO 1 .....	1
Marco teórico.....	1
1.1. Fundamentación teórica.....	1
1.1.1. Columna vertebral.....	1
1.1.2. La escoliosis.....	2
1.1.3. Clasificación de la escoliosis .....	3
1.1.4. Medición del grado de curvatura de la escoliosis .....	3
1.1.5. Exámenes médicos de la columna vertebral .....	5
1.2. Redes neuronales convolucionales .....	5
1.2.1. Inteligencia Artificial (IA) .....	5
1.2.2. Visión Artificial.....	6
1.2.3. Introducción a redes neuronales convolucionales (CNN).....	6
1.2.4. Tipos de capa.....	7
1.2.5. Modelos de aprendizaje automático.....	10
1.2.6. Clasificación, detección y segmentación.....	11
1.2.7. Mask R-CNN .....	13
1.3. Metodología de desarrollo de software .....	15
1.3.1. Proceso de descubrimiento del conocimiento (KDD) .....	16
1.4. Aplicativo .....	17
1.4.1. Detalles del sistema.....	17
1.4.2. Lenguaje de programación .....	18
1.4.3. Servidor Web .....	20
1.4.4. Base de datos .....	20
1.4.5. Angular.....	21
1.5. Trabajos relacionados.....	22
En la Medicina .....	22
En la Gastronomía.....	23
En la Conducción.....	23

CAPÍTULO 2 .....	25
Desarrollo .....	25
2.1. Visión general del proyecto.....	25
2.2. Entregables del proyecto .....	25
2.3. Organización del proyecto .....	26
2.3.1. Participantes del Proyecto .....	26
2.3.2. Roles y Responsabilidades .....	26
2.4. Gestión del Proyecto.....	26
2.4.1. Estimaciones.....	26
2.4.2. Plan del proyecto .....	28
2.5. Fase de recolección de datos .....	28
2.6. Fase de selección, preprocesamiento y transformación.....	29
2.6.1. Selección y Preprocesamiento .....	29
2.6.2. Transformación .....	30
2.7. Fase de entrenamiento de la red neuronal convolucional .....	31
2.7.1. Archivos de configuración.....	31
2.7.2. Entrenamiento.....	33
2.8. Fase de evaluación e interpretación.....	35
2.8.1. Predicciones.....	35
2.8.2. Clasificación de propuestas.....	37
2.8.3. Generación de mascararas.....	38
2.9. Algoritmo utilizando el método Cobb .....	39
2.9.1. Regresión Polinomial .....	39
2.9.2. Creación del modelo de la columna vertebral.....	40
2.9.3. Cálculo del ángulo de desviación de la columna vertebral .....	42
2.9.4. Resultados del algoritmo desarrollado.....	45
2.10. Desarrollo del Aplicativo .....	47
2.10.1. Diseño de la base de datos .....	47
2.10.2. Desarrollo Backend .....	48
2.10.3. Desarrollo Frontend .....	48
2.10.4. Presentación de resultados en el Aplicativo Web.....	48
CAPÍTULO 3 .....	54
Resultados .....	54
3.1. Análisis de resultados del entrenamiento.....	54
3.2. Muestreo.....	55
3.3. Análisis de resultados mediante la prueba t-Student.....	59
3.4. Confiabilidad del sistema .....	62
3.5. Discusión .....	63
CONCLUSIONES .....	65

RECOMENDACIONES.....	66
GLOSARIO DE TÉRMINOS .....	67
ANEXOS .....	75

## Índice de Figuras

Fig. 1. Diagrama de Ishikawa.....	XV
Fig. 2. Arquitectura del Proyecto .....	XVII
Fig. 3. Proceso Metodológico .....	XVII
Fig. 4. Columna Vertebral.....	1
Fig. 5. Escoliosis .....	2
Fig. 6. Ángulo de Cobb .....	4
Fig. 7. Redes Neuronales Convolucionales (CNN).....	7
Fig. 8. Obtención de la matriz de activación .....	8
Fig. 9. Función de activación ReLu .....	8
Fig. 10. Submuestreo con Max-Pooling de 2x2 .....	9
Fig. 11. Función Softmax.....	10
Fig. 12. Ejemplo: Clasificación de imágenes .....	11
Fig. 13. Localización de imágenes .....	11
Fig. 14. Detección de objetos .....	12
Fig. 15. Segmentación semántica .....	12
Fig. 16. Segmentación de instancias .....	12
Fig. 17. MaskR-CNN para la segmentación de instancias .....	14
Fig. 18. Extracción de características en FPN .....	14
Fig. 19. Arquitectura ResNet .....	15
Fig. 20. Red residual.....	15
Fig. 21. Fases del desarrollo del aplicativo .....	18
Fig. 22. Matriz de conceptos de proyectos relacionados.....	24
Fig. 23. Programa de visualización de radiografías.....	29
Fig. 24. Etiquetado de las imágenes .....	30
Fig. 25. Importación de la arquitectura Mask RCNN .....	31
Fig. 26. Configuración para entrenamiento .....	32
Fig. 27. Entrenar modelo .....	32
Fig. 28. Comando para entrenar el modelo .....	33
Fig. 29. Comando para reanudar el entrenamiento .....	33
Fig. 30. Configuración de uso de GPU.....	34
Fig. 31. Entrenamiento en Google Colab .....	34
Fig. 32. Anclajes y sus predicciones .....	35
Fig. 33. Refinamiento de anclas .....	36
Fig. 34. Anclajes positivos .....	36
Fig. 35. Anclajes positivos y sus predicciones .....	37

Fig. 36. Predicción del objeto .....	37
Fig. 37. Mascara del objeto .....	38
Fig. 38. Resultado final de la segmentación de instancias.....	38
Fig. 39. Curvas de aprendizaje.....	39
Fig. 40. Resultado de predicción del modelo Mask R-CNN .....	40
Fig. 41. Modelo columna vertebral mediante coordenadas.....	41
Fig. 42. Regresión polinomial con 12 grados.....	41
Fig. 43. Regresión polinomial con 5 grados .....	42
Fig. 44. Regresión polinomial con 35 grados.....	42
Fig. 45. Recta tangente y recta normal .....	43
Fig. 46. Ángulo entre dos rectas.....	44
Fig. 47. Cálculo automático del ángulo de curvatura.....	45
Fig. 48. Curvatura de tipo normal con 9.85 grados.....	45
Fig. 49. Escoliosis de tipo leve con 10.76 grados .....	46
Fig. 50. Escoliosis de tipo grave con 40.12 grados .....	46
Fig. 51. Escoliosis de tipo leve y moderada con 17.22 y 23.28 grados respectivamente.....	46
Fig. 52. Arquitectura aplicativa .....	47
Fig. 53. Diseño base de datos.....	47
Fig. 54. Inicio de sesión .....	49
Fig. 55. Agregar nuevo paciente .....	49
Fig. 56. Actualizar paciente .....	50
Fig. 57. Eliminar paciente .....	50
Fig. 58. Límite de consultas y búsqueda.....	51
Fig. 59. Resultado final con escoliosis de 41.32 grados.....	52
Fig. 60. Resultado final con escoliosis de 40.12 grados.....	52
Fig. 61. Resultado final de escoliosis con dos curvaturas de tipo grave.....	53
Fig. 62. Resultado final con escoliosis de 30 grados.....	53
Fig. 63. Pérdida Entrenamiento.....	54
Fig. 64. Pérdida Evaluación.....	55
Fig. 65. Medición Manual vs Medición Sistema.....	58
Fig. 66. Coeficiente de correlación.....	62

## Índice de Tablas

Tabla 1. Clasificación de la Escoliosis. ....	4
Tabla 2. Lenguaje DDL.....	21
Tabla 3. Lenguaje DML .....	21
Tabla 4. Entregables del Proyecto .....	25
Tabla 5. Directores de las Áreas Comprendidas .....	26
Tabla 6. Roles y Responsabilidades del proyecto .....	26
Tabla 7. Talento Humano .....	27
Tabla 8. Recursos Materiales.....	27
Tabla 9. Fases del proyecto y su distribución en horas.....	28
Tabla 10. Dataset para entrenamiento.....	29
Tabla 11. Medición del Ángulo Cobb por especialistas .....	55
Tabla 12. Grado de desviación manual y grados de desviación del sistema .....	57
Tabla 13. Prueba t para medias de dos muestras relacionadas .....	61
Tabla 14. Prueba t con dos muestras relacionadas entre observadores .....	62

## Resumen

La escoliosis es una desviación de la columna vertebral que afecta a niños, adolescentes e incluso adultos. La curvatura puede estar presente tanto en la parte lumbar como en la torácica, la escoliosis se presenta en forma C o S ocasionando problemas respiratorios y de postura. Esta investigación tiene como objetivo principal analizar imágenes radiográficas de la columna vertebral en pacientes que padecen escoliosis, para calcular de forma automática el ángulo de desviación y conseguir una optimización en el proceso que conlleva hacer esto de forma manual, reduciendo el tiempo de cálculo y el menor esfuerzo humano. Para ello se utilizó una red neuronal convolucional basada en regiones (Mask R-CNN) permitiendo obtener la zona de la columna vertebral, seguidamente se obtiene un modelo de la columna en base al centro longitudinal de la máscara del objeto y mediante regresión polinomial aplicada a 12 grados, se ajustó los datos a la curva de la columna vertebral de forma natural. Por último, se localiza el inicio y final de la curvatura para trazar rectas tangentes que al cruzarse entre sí se obtiene el ángulo de desviación. Para probar su efectividad, se realizó una comparación entre el grado de desviación del sistema y el grado de desviación manual realizado por especialistas, entre las dos muestras se obtiene un coeficiente de correlación de 0.92 % demostrando su confiabilidad.

## Abstract

Scoliosis is a deviation of the spine that affects children, adolescents and even adults. The curvature may be present in both the lumbar and thoracic portions, scoliosis presents in a C or S shape causing breathing and posture problems. The main objective of this research is to analyze radiographic images of the spine in patients suffering from scoliosis, to automatically calculate the deviation angle and get optimization on the process involved by doing this manually, reduced computation time and less human effort. To do this, a region-based convolutional neural network (Mask R-CNN) was used to obtain the area of the spine, then a model of the column is obtained based on the longitudinal center of the object mask and through polynomial regression applied to 12 degrees, the data fit the curve of the column in a natural way. Finally, the beginning and end of the curvature are located to draw tangent lines that intersect each other to obtain the angle of deviation. To test its effectiveness, a comparison was made between the degree of deviation of the system and the degree of manual deviation performed by specialists, a correlation coefficient of 0.92% is obtained between the two samples, demonstrating its reliability.

**Palabras clave:** Inteligencia artificial, redes neuronales convolucionales, columna vertebral, escoliosis, Mask RCNN, visión artificial.

# INTRODUCCIÓN

## **Problema**

### **Antecedentes**

Para obtener radiografías de la columna vertebral se utiliza los rayos X, que atraviesa los objetos como el cuerpo humano y se obtiene imágenes de los tejidos menos densos como huesos, con la ayuda de esta tecnología permite detectar anomalías o deformidades del esqueleto humano. Existe muchas áreas que se especializan en estos fenómenos, este trabajo de investigación se enfocara en pacientes que padecen escoliosis. El procedimiento para calcular el ángulo de desviación de la columna se lo realiza de forma manual con aparatos denominado escoliómetro, este instrumento fue desarrollado por Bunnell en 1984, para evaluar el grado de escoliosis, este nivel dispone de una escala numerada entre 0-30 grados, el procedimiento es poco preciso y consume tiempo en ejecutarlo. Los centros de rayos X en el cantón Cayambe cuentan con sistemas de medición digital que es operado por especialistas y pueden realizar mediciones trazado líneas perpendiculares en la curvatura de la columna en imágenes radiográficas. El ángulo de Cobb es el método más utilizado para calcular el ángulo de desviación, fue creado por Robert John Cobb. Para obtener el ángulo de Cobb se debe localizar las vértebras iniciales y finales de la deformidad o curva, a continuación, se trazan líneas perpendiculares que al intersecarse o cruzarse entre sí, se obtendrá el ángulo de desviación.

### **Situación Actual**

Actualmente los centros de salud especializados en detectar problemas de la columna vertebral en el cantón Cayambe, no cuentan con sistemas informáticos que facilite calcular el ángulo de desviación de la columna vertebral de forma automática, los procedimientos lo realizan con aparatos manuales o sistemas digitales que requiere de un experto. No hay información de software especializado en escoliosis para calcular la curvatura de la columna en imágenes radiográficas.

## Prospectiva

La presente investigación permite reducir el tiempo de cálculo de ángulo de deformidad de la columna vertebral en pacientes que padecen escoliosis, los resultados serán almacenados en el sistema para verificar su evolución en los distintos tratamientos aplicados. Los centros de salud especializados en la columna vertebral en el cantón Cayambe, guardan sus registros en hojas físicas o sistemas digitales limitados, el gran problema que genera esto, es la inconsistencia de datos, pérdida de información, datos duplicados, etc. Estos centros de salud no cuentan con un software enfocado al área de escoliosis, su enfoque es a nivel general, provocando que la información del paciente no sea recolectada de forma detallada por el sistema implantado.

## Planteamiento del Problema

El procedimiento para calcular el ángulo de desviación de la columna vertebral se lo realiza de forma manual mediante aparatos o sistemas digitales, consume tiempo en obtener los resultados y diagnosticar su tratamiento. No hay mucha información sobre sistemas informáticos especializados en escoliosis que se utilicen en centros de salud del Cantón Cayambe. Por ello, el presente trabajo pretende optimizar el procedimiento de detección del ángulo de curvatura de la columna en pacientes que padecen escoliosis, mediante técnicas de inteligencia artificial y visión por computador en imágenes de rayo x.

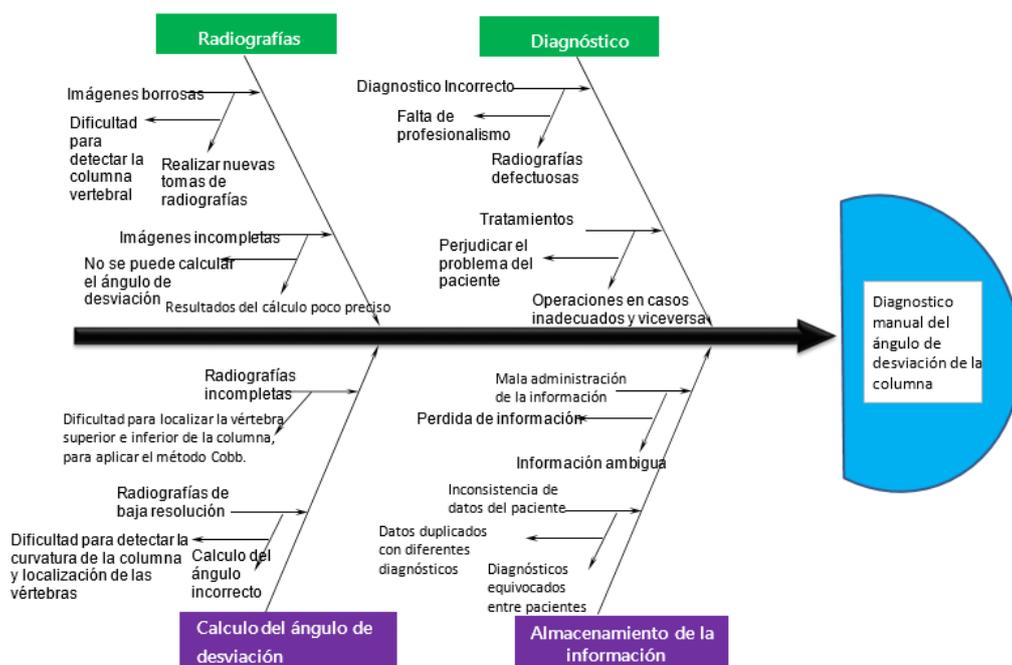


Fig. 1. Diagrama de Ishikawa.

## **Objetivos**

### **Objetivo General**

Analizar imágenes de radiografías de la columna vertebral para el cálculo del ángulo de desviación en pacientes que padecen escoliosis utilizando técnicas de Inteligencia Artificial.

### **Objetivos Específicos**

- Elaborar un marco teórico referencial respecto a la detección de escoliosis utilizando redes neuronales profundas para calcular la curvatura de la columna vertebral.
- Desarrollar un algoritmo aplicando el método de Cobb para calcular el ángulo de desviación de la columna vertebral utilizando Python y el Framework Tensorflow.
- Validar los resultados obtenidos con especialistas en el área de escoliosis.

### **Alcance**

El desarrollo de la aplicación web para analizar imágenes de radiografías de la columna vertebral y calcular el ángulo de desviación en pacientes que padecen escoliosis, y a su vez definir el grado de gravedad, como leve, mediana o grave, los resultados serán almacenados en el sistema para verificar su evolución de acuerdo con el tratamiento que el especialista lo recomiende. Con el dataset proporcionadas por el Centro de rayos X en el Cantón Cayambe, se entrenará una red neuronal convolucional basadas en región (Mask R-CNN) para obtener la zona de interés y con ello desarrollar un algoritmo que calcule el ángulo de desviación aplicando el método Cobb. El aplicativo se desarrollará en el Framework Angular como parte del Front-End y en la parte del Back-End se utilizará los lenguajes de programación JavaScript y Python, la base de datos que se implementará será Postgresql. Para el análisis de imágenes se utilizará las librerías de Keras y Tensorflow proporcionadas por el lenguaje de Python, todo esto se desarrollará en el sistema operativo Linux. Los resultados obtenidos serán evaluados con especialistas en el área de escoliosis y se obtendrá el grado de confiabilidad del sistema.

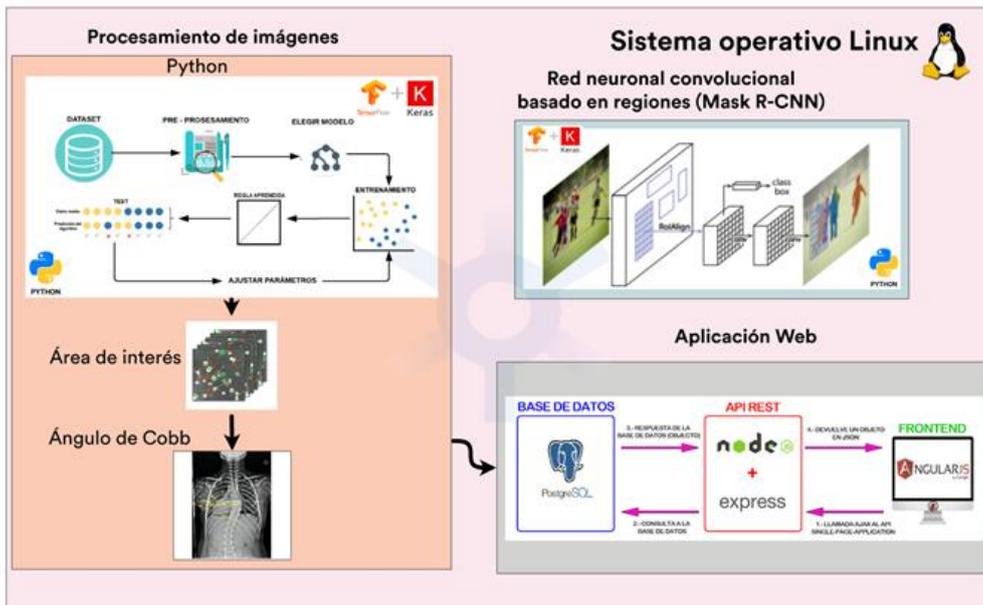


Fig. 2. Arquitectura del Proyecto

## Metodología

Para el desarrollo de la presente propuesta, se aplicará la investigación de campo y descriptiva, con respecto a la investigación de campo, se busca los datos en centros de rayos X especializados en tomas de la columna vertebral, y la investigación descriptiva es debido al estudio de patologías de pacientes que padecen escoliosis, teniendo mayor enfoque en dar a conocer la situación de gravedad al calcular el ángulo de desviación de la columna vertebral.

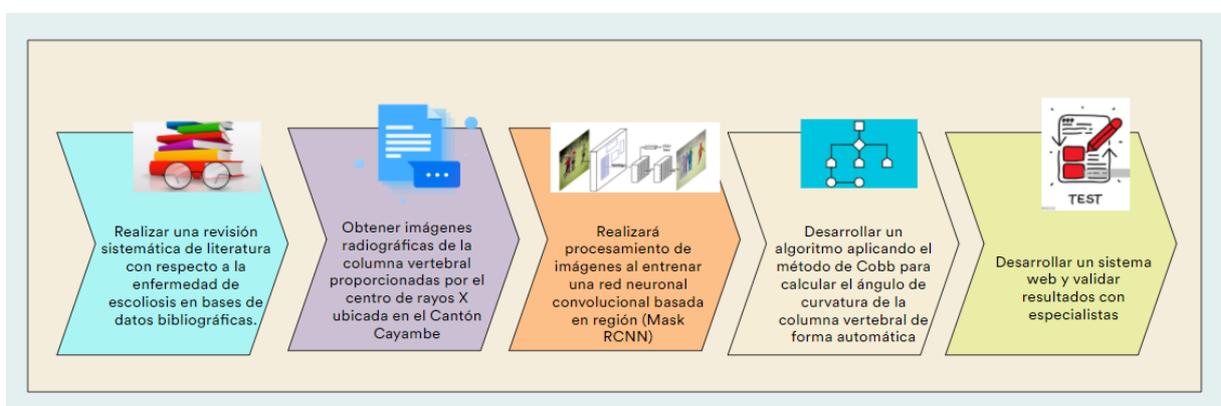


Fig. 3. Proceso Metodológico

## **Justificación**

En la actualidad, es necesario que la información sea procesada y almacenada de forma eficiente con la ayuda de herramientas tecnológicas que agiliza el proceso, la incorporación de inteligencia artificial en los sistemas informáticos ha permitido resolver problemas de razonamiento humano, un gran ejemplo es la IA en procesamiento de imágenes para detectar objetos con gran precisión y realizar una clasificación de los mismos.

El presente proyecto pretende agilizar el proceso de cálculo del ángulo de curvatura de la columna vertebral utilizando técnicas de IA y visión por computador, mediante un sistema web permitirá procesar imágenes radiografías de rayos X y gestionar la información con los resultados obtenidos, ejecutando las actividades con el menor esfuerzo humano y en el menor tiempo posible.

Esta propuesta de investigación tiene relación con el objetivo 3 de la ODS donde expresa “Garantizar una vida sana y promover el bienestar para todos en todas las edades”. El proyecto se identifica con este objetivo, porque el sistema web pretende mejorar la calidad de vida de las personas. Por lo general los niños son los más propensos a presentar síntoma de escoliosis a diferencia de los adultos, por ello, el sistema puede ser aplicado para prevenir alteraciones posturales en personas que presentan síntomas de escoliosis y así garantizar una vida sana y libre de problemas de la columna vertebral, agilizando el diagnóstico y tratamiento temprano de la enfermedad.

**Tecnológica:** Los lenguajes de programación JavaScript y Python tiene licencia abierta y puede ser usando en cualquier escenario. Con respecto a la Inteligencia Artificial, Python es una buena opción porque cuenta con gran número de librerías capaces de agilizar el proceso de desarrollo. Una de las librerías más importantes es TensorFlow posicionado como la herramienta líder en el sector de Deep Learning

# CAPÍTULO 1

## Marco teórico

### 1.1. Fundamentación teórica

#### 1.1.1. Columna vertebral

La columna vertebral es la estructura orientadora del equilibrio del cuerpo humano, se compone de elementos óseos superpuestos denominados vértebras, la columna vertebral consta de un variado grupo de músculos que permiten la movilidad, siendo resistentes y flexibles. En este grupo se encuentra los músculos extensores, flexores y oblicuos, los músculos extensores se encuentran unidos a la parte posterior de la columna vertebral, mientras que los músculos flexores están unidos en la parte anterior de la columna vertebral, permitiendo en conjunto estar de pie y levantar objetos, los músculos oblicuos están unidos a la columna vertebral en las partes laterales, ayudan a rotar la columna y mantener la postura. (Lisintuña Molina, 2015).

La columna vertebral tiene tres funciones principales, proteger la médula espinal, las raíces nerviosas y los distintos órganos internos del cuerpo humano.

#### Anatomía de la columna vertebral.

La columna vertebral está constituida por vértebras, que son 33 o 34 elementos óseos, discordeos que se superponen, distribuidas de la siguiente manera: 7 cervicales, 12 dorsales, 5 lumbares, 9 o 10 pélvicas. Son denominados vértebras independientes a las cervicales, dorsales y lumbares, mientras que el sacro y el cóccix son dos elementos unidos, se encuentra en la parte pélvica (González Acevedo & Rodríguez Pérez, 2016).

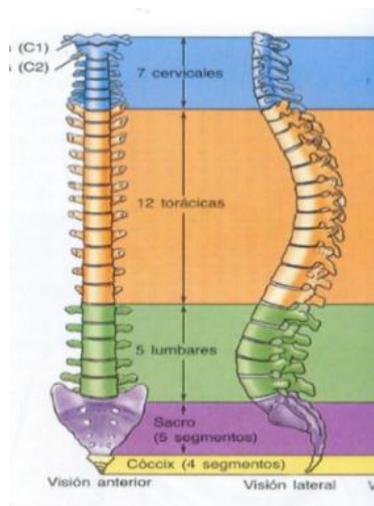


Fig. 4. Columna Vertebral.  
Fuente: (Miranda, 2017)

## Región Cervical

Se encuentran ubicados en el cuello y posee mayor movilidad, consta de 7 vertebras definidas de C1 a C7 contando de arriba hacia abajo, Estas vertebras protegen el tallo cerebral y la médula espinal, sostiene el peso del cráneo que está sujeta por C1 llamado (Atlas) y en conjunto con C2 (Axis) permiten que la cabeza gire (Fernández Ávalos, 2011).

## Región Torácica o Dorsal

Consta de 12 vértebras desde T1 a T12, es más resistente y posee mayor estabilidad que la región cervical, pero su movilidad es menor que en la parte cervical (Juan et al., 2018).

## Región Lumbar

Posee un grupo de 5 grandes vértebras desde L1 a L5, permitiendo tener mayor movilidad que el segmento torácico, esta región está diseñada para cargar la mayor parte del peso corporal (Chacón Bustamante & Luna Yela, 2013).

## Región Sacra

El peso del cuerpo es transmitido por la región sacra hacia los miembros inferiores, con los huesos de la cintura forman un anillo sólido (Oliveira et al., 2011).

### 1.1.2. La escoliosis

La escoliosis es una desviación o curvatura de la columna vertebral en forma de C o S, esta desviación puede producirse hacia la derecha o izquierda de la columna vertebral, por lo general ocasiona daños a los órganos, especialmente los pulmones, puede afectar a la caja torácica porque la rotación vertebral lleva a las articulaciones costovertebrales en su deformación. El 90% de las escoliosis juveniles se desconoce su origen, a esta parte se denomina escoliosis idiopática y afecta a cualquier parte de la columna vertebral (Lisintuña Molina, 2015).



Fig. 5. Escoliosis  
Fuente: (SBV ORTOPEDIA, 2019)

La escoliosis ocurre con mayor frecuencia durante el periodo de crecimiento anterior a la pubertad y puede ser hereditario si algún familiar contiene esta patología, pero la mayoría de los casos los niños que lo padecen no tienen antecedentes familiares de la enfermedad. Los síntomas de la escoliosis pueden ser por hombros desparejos, cintura despareja, un omóplato es más elevado que el otro.

### 1.1.3. Clasificación de la escoliosis

La sociedad Americana de la columna vertebral “Scoliosis Research Society” (1996) ha clasificado a la escoliosis en 4 partes (Gacitúa M et al., 2016).

- **Escoliosis idiopática:** Es la más común de todas y puede presentar dos curvaturas en la columna vertebral, tiene un origen desconocido de su causa.
- **Escoliosis neuromuscular:** Es la enfermedad que afecta a la médula espinal, a los músculos, como también al cerebro provocando una parálisis cerebral.
- **Escoliosis degenerativa:** Se origina cuando colapsa traumáticamente alguna de las vértebras, por ejemplo, de una lesión causada en una cirugía o enfermedad.
- **Escoliosis congénita:** Causada por una anomalía ósea, se presenta en el nacimiento.

### 1.1.4. Medición del grado de curvatura de la escoliosis

La escoliosis se mide en grados y cuenta con diferentes métodos para calcular el ángulo de desviación de la columna vertebral, como el método de NSH\_moe, método de Ferguson, el método de Cobb, entre otros. El método de Cobb es el más utilizado y efectivo para validar una escoliosis, permite cuantificar la magnitud de la deformación y la gravedad de la columna vertebral, al obtener los resultados se podría evitar una cirugía innecesaria y elegir el mejor tratamiento a seguir (Lisintuña Molina, 2015).

En este trabajo de investigación se implementará el método de Cobb por ser efectivo para valorar una escoliosis y sobre todo al ser el método más utilizado en medir el ángulo de curvatura. Se utilizarán imágenes radiográficas frontales de la columna vertebral.

Para medir el ángulo de Cobb se debe encontrar la vértebra inicial y final del comienzo de la deformidad de la curvatura, para ello se debe localizar el ápex denominado a la vértebra con la mayor inclinación de la curvatura desde un eje central o línea plomada, trazada en el centro C7 o línea vertical en el centro sacro, esto permite identificar la inclinación de la curvatura hacia el lado derecho o izquierdo. A continuación, se debe elegir las vértebras de ambos extremos de la curvatura, esta selección será las vértebras que tienen la mayor inclinación hacia el lado identificado anteriormente. Por último, se trazan líneas

perpendiculares ubicadas en el límite superior de la vértebra inicial y el límite inferior de la vértebra final de la curvatura, el punto donde se intersecan o se cruzan entre sí las líneas, se obtendrá el ángulo de desviación (Gacitúa M et al., 2016).

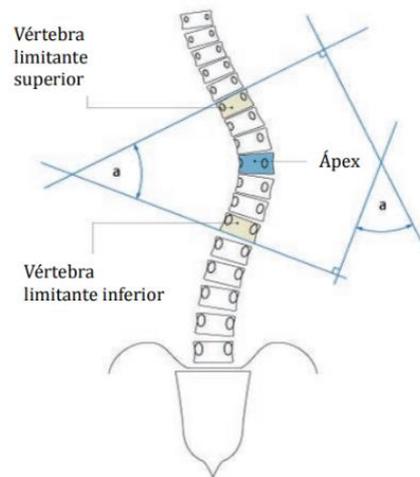


Fig. 6. Ángulo de Cobb  
Fuente: (Sánchez Márquez, 2016)

El rango del ángulo de Cobb va desde las mediciones de 10 hasta los 120 grados (Blanch Alonso, 2021).

- a) **Menor de 10°:** Se considera una columna vertebral normal, aunque presente una pequeña curvatura.
- b) **Igual a 10°:** A partir desde este rango se afirma que la persona tiene comienzos de escoliosis y se denomina leve cuando no llega a los 20 grados.
- c) **Entre 20° y 40°:** Se considera moderada entre los 20 y 40 grados, el paciente debe usar soporte ortopédico.
- d) **Mayor de 40°:** Es considerada grave porque la curvatura de la columna vertebral es muy profunda y ningún tratamiento detiene su evolución, en esta situación se recomienda cirugía.

Tabla 1. Clasificación de la Escoliosis.

Grado de inclinación	Tipos de escoliosis
< 10	Normal
10° - 19°	Leve
20° - 40°	Moderada
> 40°	Grave

### **1.1.5. Exámenes médicos de la columna vertebral**

Las imágenes médicas obtenidas por rayos x es un gran instrumento para la práctica clínica, porque permite detectar patologías en nuestro sistema óseo, las imágenes obtenidas permiten realizar una investigación en el proceso previo a la cirugía o encontrar otras alternativas curativas ante una patología.

Cuando el paciente padece de escoliosis más severo, se realiza tomas de imágenes radiográficas en diferentes posiciones, es fundamental que las radiografías muestren claramente de forma global y regional la alineación de la columna vertebral, para ello se realiza proyecciones postero-anteriores (frontal) y lateral. La proyección postero-anteriores permita identificar la inclinación de la curvatura y la proyección lateral permita saber si la curvatura es estructural y hasta que grado se puede corregir (Sánchez García & Méndez Durán, 2015).

Los pacientes que se han realizado un examen radiológico son expuestos a una radiación mínima, la exposición de radiación depende del tipo de estudio que una persona se somete, también depende del estado de la persona como su edad, sexo, cuerpo, entre otros factores. Es difícil saber que tan expuesto a las radiaciones mínimas en una persona este en riesgo de contraer cáncer. Si la dosis es muy alta, puede tener efectos agudos como quemaduras por radiación, enrojecimiento de la piel, pérdida de cabello, entre otros. Es importante mencionar que se debe proteger a los niños de la radiación como sea posible, porque su cuerpo es más sensible (Alteri & Kalidas, 2016).

## **1.2. Redes neuronales convolucionales**

### **1.2.1. Inteligencia Artificial (IA)**

La inteligencia artificial es un sistema que en cierto grado tiene la posibilidad de autoaprendizaje a través del análisis de datos, su aprendizaje específico tiene la capacidad de replicar o imitar un razonamiento humano (Brevity, 2021).

La IA cuenta con la capacidad de procesar datos y tomar decisiones a base de un entrenamiento profundo. Los algoritmos de IA son usados para resolver tareas específicas y complejas, por ejemplo, en reconocer tus gustos de videos y sugerirte el próximo con el objetivo de llamar tu atención evitando el aburrimiento. Un ejemplo muy grande en visión artificial es la conducción autónoma en vehículos eléctricos, su arquitectura de red neuronal reconoce cientos de objetos en tiempo real y esta información es utilizada para una conducción autónoma más precisa.

### **1.2.2. Visión Artificial**

La visión artificial es un campo de la IA, su objetivo es analizar imágenes digitales y videos, y mediante redes neuronales convolucionales extraer información, previo al etiquetado y al proceso de entrenamiento la red neuronal reconoce objetos y objetos superpuestos con gran precisión. La visión artificial es muy similar a la visión humana, incluso pueden ser mucho más rápido que el ojo humano, siendo capaz de procesar más de 100 fotogramas por segundo gracias al poder computacional actualmente (IBM, 2022).

La visión artificial utiliza dos tecnologías para su funcionamiento, entre ellos se encuentra el machine learning y una red neuronal convolucional, Al juntar estas tecnologías la CNN distinguen imágenes al igual que el ojo humano. Se ha utilizado CNN en la medicina identificando patologías en imágenes de rayos x como cáncer de pulmón, obtenido al final un resultado con alto porcentaje de predicción.

### **1.2.3. Introducción a redes neuronales convolucionales (CNN)**

Las redes neuronales convolucionales (“Convolutional Neural Network” o CNN) es una red neuronal multicapa diseñada específicamente para visión artificial. Dentro de su estructura usa capas de convolución para recibir y procesar datos de imágenes, las primeras capas se encargan de comprimir y detectar rasgos característicos de un objeto y mientras se adentra en las capas más profundas ya puede reconocer formas completas del objeto, hasta que la capa final clasificará y dará una predicción en base a lo aprendido en el entrenamiento. Las CNN se inspiran en el funcionamiento del cerebro humano y se puede utilizar de innumerables formas, como la clasificación de imágenes. Por ejemplo, clasificar entre perros y gatos (García Sánchez, 2019).

La CNN recibe como entrada una imagen, esta imagen está formado de píxeles que cada una de ellas está representada por un número entre 0 y 255, esta es la forma que una computadora trabaja con imágenes. La estructura de una CNN cuenta con dos partes muy importantes, como primer punto se hace la extracción de características de la imagen y el segundo punto se realiza la clasificación de la imagen. La red neuronal convolucional cuenta con 4 diferentes capas: capa convolucional, capa relu, capa pooling, capas completamente conectadas. Las tres primeras capas pertenecen a la extracción de características de la imagen y la última capa pertenece a la clasificación. A continuación, se explicará con mayor detalle el funcionamiento de cada capa.

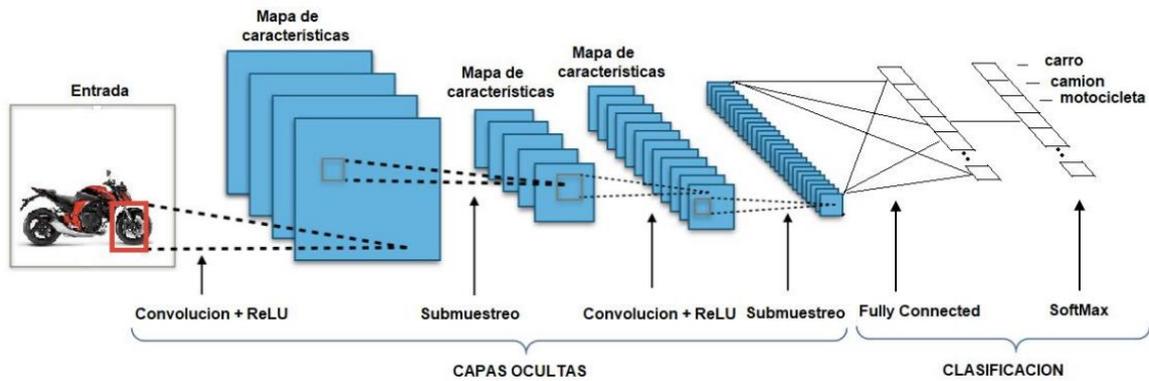


Fig. 7. Redes Neuronales Convolucionales (CNN)  
Fuente: (Hernán Mondragón et al., 2015)

#### 1.2.4. Tipos de capa

##### Capas convolucionales (Convolutional layers)

Las primeras capas que se conectan al tener como entrada una imagen son las capas convolucionales, su objetivo es extraer los rasgos característicos de la imagen, lo realiza aplicando filtros denominados kernels. De esta manera, cada neurona oculta se conecta con un subconjunto pequeño de neuronas de la primera capa de entrada. Con respecto a la convolución es una operación de productos y sumas entre la imagen de entrada y el filtro (kernel). El kernel es una matriz que se desliza por todos los píxeles de la imagen, en cada posición la multiplicación por elementos es calculado y se genera un elemento de matriz de salida, almacenando el resultado en la siguiente posición de la matriz de activación, este proceso se repite a lo largo de toda la imagen desplazándose de izquierda a derecha (Landa, 2018).

Las capas convolucionales ayudan a reducir el número de neuronas que conforma la red, permitiendo detectar rasgos característicos de una imagen al momento de entrenar la red. También, las capas convolucionales reducen el procesamiento computacional y la complejidad del sistema.

Un ejemplo de una matriz de entrada 7x7 y un filtro de 3x3, para obtener la dimensión de la matriz de activación se realiza con la siguiente fórmula:  $[\text{ancho original} - \text{ancho kernel} + 1] [\text{alto original} - \text{alto kernel} + 1]$ , reemplazando los valores en la fórmula se obtiene el siguiente resultado:  $[7-3+1] [7-3+1] = [5][5]$ .

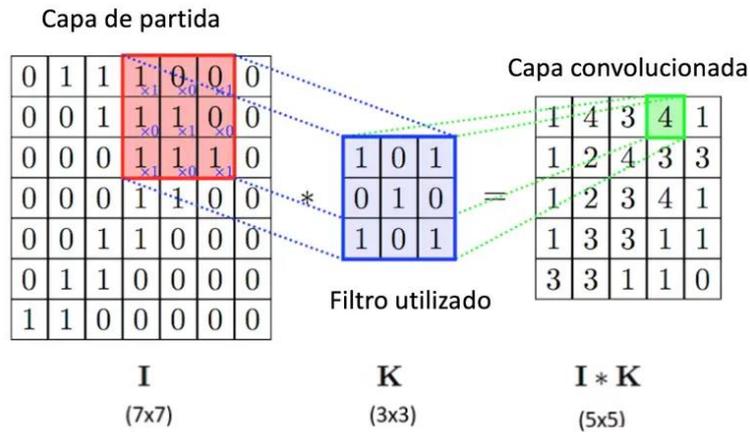


Fig. 8. Obtención de la matriz de activación  
Fuente: (Calvo, 2017)

### Capas ReLu (ReLU layers)

ReLU es una función de activación más utilizada en las capas ocultas en redes neuronales convolucionales (CNN). Los datos reales que maneja al momento de entrenar una CNN son en su mayoría no lineales. Hay muchas funciones de activación, entre ellas encontramos la función sigmoide utilizada para predecir la probabilidad de un resultado donde la clasificación es 0 y 1. La función de activación ReLu es usada en las capas convolucionales. La salida de ReLu está definida por una función:  $\text{ReLU}(x) = \max(0, x)$ , aplicando esta función los valores de entrada después de haber pasado por la capa de convolución, se obtiene una matriz con valores negativos y la función ReLu retorna una matriz donde los valores negativos son remplazados por 0 (Landa, 2018).

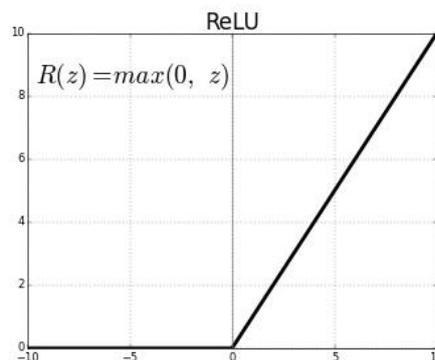


Fig. 9. Función de activación ReLu  
Fuente: (Freire & Silva, 2019)

La importancia de utilizar la función de activación ReLu es porque los valores negativos no son importantes en el procesamiento de imágenes, por ello se establece en 0. Los valores positivos no son afectados por este cambio, siguen conservando sus relaciones neuronales donde guardar algún patrón o característica de la imagen.

## Capas pooling (Pooling layers)

Al terminar la capa de convolución se encuentra la capa de pooling, se encarga de hacer una simplificación de la información obtenida por las capas convolucionales. Para realizar esta labor, es buscar que zona de la imagen contiene rasgos predominantes, para ello se utiliza un nuevo kernel con el propósito de condensar la información. Es importante destacar que existe dos tipos de pooling: mean-pooling y max-pooling, en esta investigación se explicará el funcionamiento de max-pooling. Continuando con el nuevo kernel o filtro, en este ejemplo se utilizará una dimensión de 2x2, se encarga de encontrar el máximo valor al recorrer el filtro por toda la matriz con pasos de dos. La matriz resultante es considerablemente menor que la matriz de entrada (Gutiérrez, 2021).

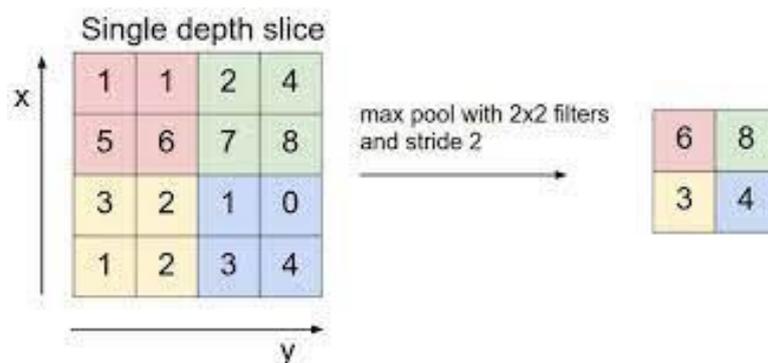


Fig. 10. Submuestreo con Max-Pooling de 2x2  
Fuente:(DataScientest, 2020)

El objetivo de esta capa es disminuir el tamaño de conexiones entre neuronas y disminuir aún más el proceso computacional del sistema.

## Capas completamente conectadas (Fully connected layers)

Esta es la última capa del esquema de las redes neuronales convolucionales y se encargará de realizar la parte de clasificación que determinará a que clase pertenece la imagen de entrada. Esta capa estará a continuación de la última capa de pooling, para que se convierta en un tensor (un tensor de 1D es un vector, de 2D una matriz, de 3D un cubo, etc.) de una solo dimensión y que estén densamente conectadas, se lo realiza mediante la capa de "flatten". La capa "flatten" convierte en un vector teniendo como entrada una matriz, este vector representa con cada valor la probabilidad de características que pertenezca a dicha clase. Para obtener el mismo número de neuronas que el número de clases de clasificación, se puede aplicar otra capa completamente conectada, las capas completamente conectadas a veces también se denominan capas densas (Durán Suárez, 2017).

Por último para obtener el resultado de clasificación se utiliza la función de Softmax, esta función calcula la distribución de probabilidad del evento sobre el número de clases.

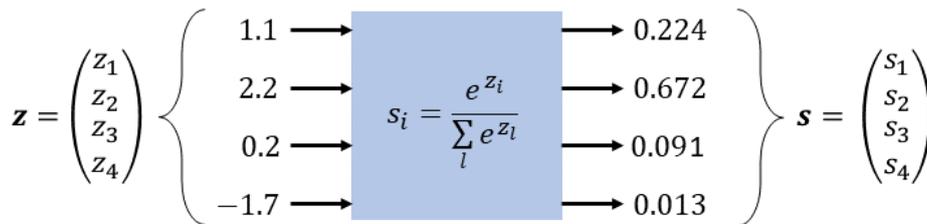


Fig. 11. Función Softmax  
Fuente: (Kurbiel, 2021)

### 1.2.5. Modelos de aprendizaje automático

La inteligencia artificial cuenta con varios métodos de aprendizaje, estos métodos tienen la capacidad de aprender con la ayuda de grandes volúmenes de información, permitiendo que el algoritmo encuentre varios patrones en el proceso de entrenamiento. Aplicando estos modelos de aprendizaje, el desarrollador no tendrá que sentarse a programar por horas tomando en cuenta todos los escenarios posibles.

Dentro del aprendizaje automático pueden aprender de 4 formas distintas, entre ellas encontramos:

- **Aprendizaje supervisado:** Estos modelos requieren de empleo de mano humana en el proceso de etiquetado. Para que el algoritmo aprenda, el modelo necesita el valor de entrada y de salida. Así a futuro el algoritmo pueda realizar predicciones conociendo primero las características (Recuero de los Santos, 2021).
- **Aprendizaje no supervisado:** Estos modelos no requieren datos etiquetados, solo necesitan como datos de entrada características de dicha situación, el objetivo de este modelo es modelizar la estructura o distribución de los datos para asumir que los datos pertenezcan a diferentes grupos clasificados (Gonzalo, 2018).
- **Aprendizaje semisupervisado:** Este modelo se balancea de forma intermedia entre el aprendizaje supervisado y no supervisado. Para entrenar el modelo puede haber datos etiquetados o no etiquetados con respecto a su salida, por ello al ingresar datos de entrada queremos obtener su dato de salida (Gonzalo, 2018).
- **Aprendizaje reforzado:** El aprendizaje por refuerzo tenemos el modelo de aprendizaje no supervisado que solo reciben re-alimentación o refuerzos, por ejemplo (premio o penalización). Su objetivo es aprender a mapear situaciones de acciones con respecto a la función de recompensa para maximizar su aprendizaje en dicha labor (Bagnato, 2020).

### 1.2.6. Clasificación, detección y segmentación

El campo de visión por computador cuenta con diferentes técnicas para extraer información de una imagen analizada. La CNN tiene un buen rendimiento de clasificación, resolviendo el problema de reconocimiento de imágenes, a medida que el rendimiento ha ido mejorando los investigadores han extendido CNN, incluyendo las tareas de detección de objetos y segmentación de imágenes. A continuación, se explica cada una de ellos con mayor detalle.

- a) **Clasificación de imágenes:** La CNN recibe un mapa de atributos de entrada, denominados como dataset, el dataset es una colección de imágenes ya previamente preparadas y etiquetadas con N clases diferentes. Como salida devuelve el porcentaje de efectividad de cada clase encontrada en la imagen de entrada (Torres Alonso, 2020).



Fig. 12. Ejemplo: Clasificación de imágenes

- b) **Localización de imágenes:** Busca en qué punto se sitúa los objetos que pertenecen a la misma clase y mediante un cuadro delimitador lo señala.

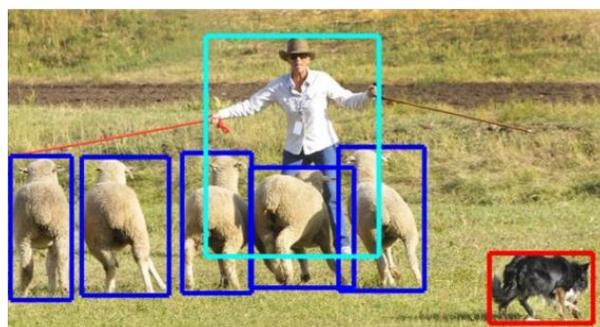


Fig. 13. Localización de imágenes

- c) **Detección de objetos:** De igual manera localiza los objetos, pero a diferencia, cada N clases etiquetadas devuelve un cuadro delimitador con la información de la etiqueta, especificando a que clase pertenece y el color de la etiqueta perteneciente (Sánchez Alor, 2020).

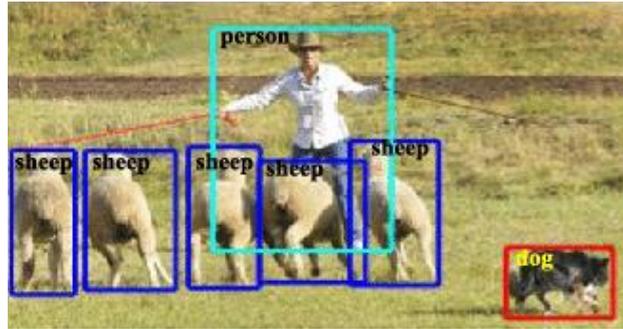


Fig. 14. Detección de objetos

d) **Segmentación semántica:** Consiste en otorgar una etiqueta o categoría a cada píxel de una imagen, el objetivo es delimitar los objetos de cada clase a nivel de píxeles otorgando la forma del objeto etiquetado (Landa, 2018).

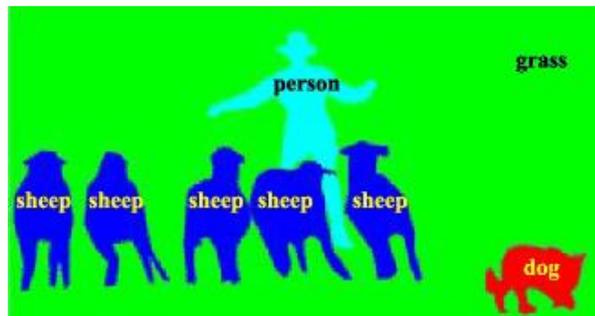


Fig. 15. Segmentación semántica  
Fuente: (Creative Commons, 2017)

e) **Segmentación de instancias:** Es la detección de objetos con segmentación semántica, la diferencia es que la segmentación de instancias detecta y etiqueta los objetos del mismo tipo o de diferente clase de manera superpuesta, siendo mucho más precisas en su detección (Torres Alonso, 2020).

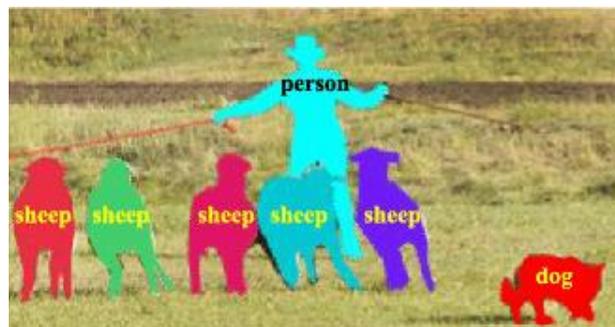


Fig. 16. Segmentación de instancias  
Fuente:(Creative Commons, 2017)

La red neuronal convolucional más utilizada en detectar objetos en una imagen es el modelo YOLO y para la segmentación de instancias se ocupa el modelo COCO. A continuación, se describe cada una de ellas.

## **YOLO**

YOLO es una red neuronal convolucional muy rápida en detectar múltiples objetos en una imagen, cada objeto es localizada mediante un cuadro delimitador donde se especifica el id de la clase y su porcentaje de predicción. El modelo detecta objetos en tiempo real adecuado para videos, si se ocupa la GPU ejecutara 45 cuadros por segundo. YOLO detecta objetos a partir de rejillas divididas en toda la imagen y realiza predicción por cada una de ellas, las probabilidades más altas serán seleccionadas para reconocer el objeto etiquetado (Raneros, 2021).

## **COCO**

Es un modelo pre-entrenado con un dataset de 330 mil imágenes y reconoce 80 tipos de objetos, el modelo cumple con la función de clasificación y detección de objetos en una imagen de entrada, cuenta con una detección más precisa y su tasa de error es mucho más pequeña que de modelos anteriores. COCO es un modelo escalable, permite agregar nuevos objetos para el previo entrenamiento, no es necesario entrenar el modelo con un dataset que contiene miles de imágenes, con cientos de imágenes es suficiente. De igual manera, es suficiente ejecutar el entrenamiento con cientos de épocas. La CNN ocupa mucho proceso computacional, por ello es recomendable ejecutar el proceso en la GPU (Srivastava et al., 2020).

### **1.2.7. Mask R-CNN**

Mask R-CNN es un algoritmo de segmentación de instancias que puede realizar numerosas tareas como clasificación de objetos, detección de objetos, segmentación semántica, segmentación de instancias y reconocimiento de gestos humanos. El modelo Mask R-CNN genera cuadros delimitadores y máscara de segmentación para cada instancia de cada objeto encontrada en la imagen, utiliza el modelo basado en red de pirámide de funciones (FPN) y una red neuronal ResNet101 (He Kaiming et al., 2018).

En la Fig. 17 se muestra el marco Mask R-CNN para la segmentación de instancias, el sistema es sorprendentemente simple, flexible y rápido, un modelo de instancias de última generación.

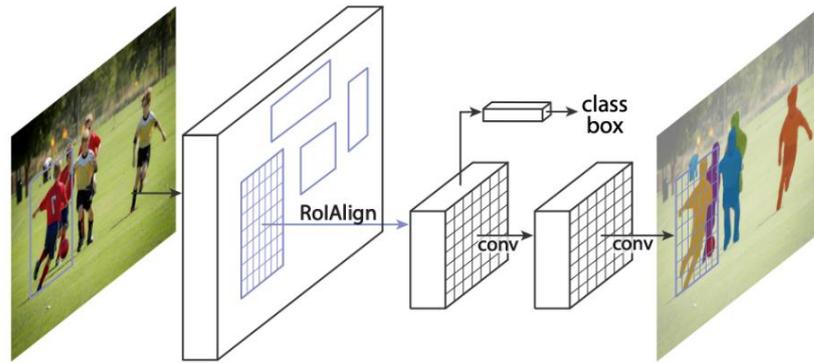


Fig. 17. MaskR-CNN para la segmentación de instancias  
Fuente: (He Kaiming et al., 2018)

### FPN (feature pyramid networks)

La principal característica de redes piramidales (FPN) es detectar objetos en diferentes escalas mediante un extractor de funciones usando una conexión lateral de abajo hacia arriba y uno de arriba hacia abajo, en todas las escalas se construye una característica semántica de alto nivel. La vía de abajo hacia arriba es la red convolucional para la extracción de características y a medida que va subiendo, la resolución espacial disminuye. La capa inferior no detecta objetos, solo la capa superior lo realiza, cuando un objeto es demasiado pequeño su detección será mucho peor en la capa superior (Lin et al., 2017).

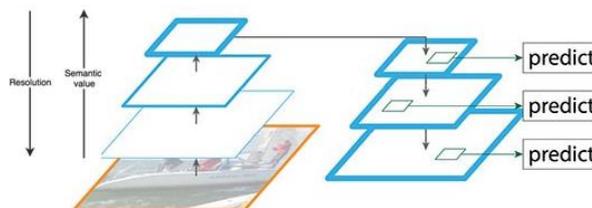


Fig. 18. Extracción de características en FPN  
Fuente: (Lin et al., 2017)

Las capas construidas anteriormente son semánticamente fuertes, pero la localización de los objetos no es tan precisa después de todo el submuestreo y reducción de resolución, para ello se agrega conexiones laterales entre las capas construidas para una mayor detección y predicción del objeto. Aplicando estas conexiones de salto facilita el entrenamiento al obtener una estructura sumamente conectada.

### ResNet101

Este modelo de Red Neuronal Convolutiva fue introducido por Microsoft en 2015 y fue ganadora del concurso al contar con una tasa de error bajo de 3.57%. Lo consiguió con los bloques residuales y las 152 capas. Con esta arquitectura nueva consiguió el mayor puntaje en clasificación, localización y detección de objetos (He et al., 2015).

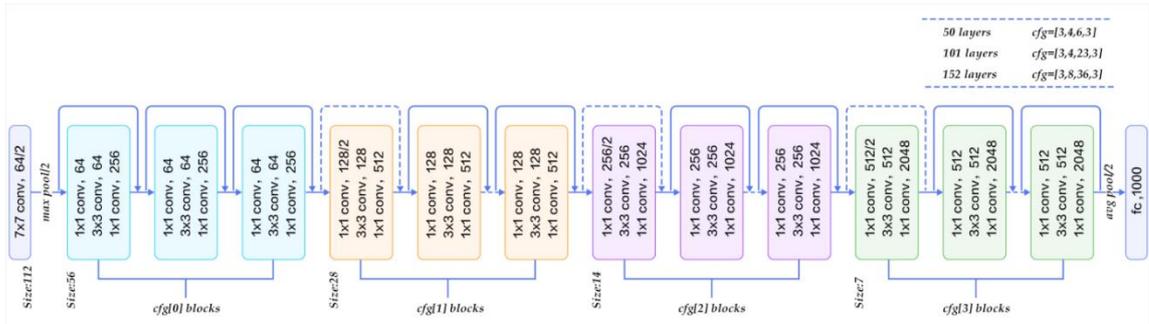


Fig. 19. Arquitectura ResNet  
Fuente: (Utrera Bursal, 2018)

ResNet cuenta con gran cantidad de capas ocultas, pero se ha demostrado que a cierta profundidad su efectividad no es la mejor a comparación con modelos de capas ocultas con menor profundidad, el problema es el desvanecimiento del gradiente (vanishing gradient) y la dimensionalidad (curse of dimensionality). A partir de este problema nace la idea de los bloques residuales, esta arquitectura se basa en dividir una red en fragmentos de tres capas y pasar la entrada a cada fragmento directamente a la siguiente capa, a esto se denomina conexión residual y los resultados afirman que tiene un mejor proceso de aprendizaje (Moreno Díaz, 2020).

A continuación, se presenta el modelo de bloque residual en la siguiente figura.

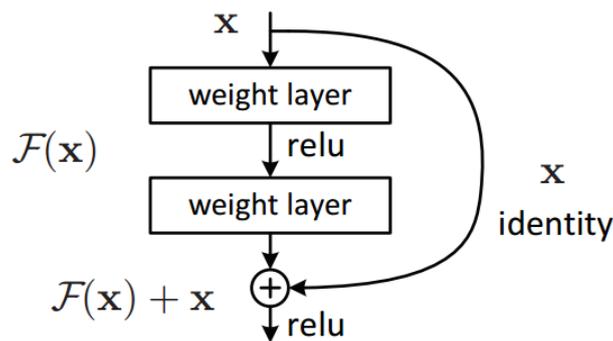


Fig. 20. Red residual  
Fuente: (He et al., 2015)

### 1.3. Metodología de desarrollo de software

Una metodología de desarrollo de software es definida como una guía con el objetivo de mantener de forma sucesiva un proceso de desarrollo, estas metodologías permiten gestionar un proyecto de software a partir de un conjunto de tareas organizadas que hace énfasis en la planificación, análisis, diseño, desarrollo, implementación, pruebas y mantenimiento (Rathnayaka & Kumara, 2020).

En el desarrollo de un proyecto de investigación es importante usar metodologías que ayuden a procesar y extraer información mediante una serie de pasos, para esta investigación se implementara la metodología KDD.

### **1.3.1. Proceso de descubrimiento del conocimiento (KDD)**

La metodología KDD del inglés Knowledge Discovery in Databases es un proceso de análisis con el fin de extraer patrones de información mediante reglas o funciones de grandes cantidades de datos, su proceso es en base a la minería de datos (data mining) para luego evaluar e interpretar los patrones del modelo que contiene el conocimiento para la toma de decisiones (Timarán Pereira et al., 2016).

Las etapas del proceso KDD son las siguientes:

#### **Fase de integración y recopilación**

En esta fase hay que tener muy claro los objetivos y límites de nuestro alcance en la búsqueda de imágenes para la construcción del dataset, es importante hacer énfasis a lo que está a nuestra disposición para no caer en un espacio infinito de datos en la que no podamos controlar.

#### **Fase de selección, preprocesamiento y transformación**

A partir del conjunto de imágenes recolectadas, se realiza las siguientes etapas para un formato integro.

- **Selección**

En esta etapa se procede a seleccionar las imágenes acordes al formato planteado en la investigación, los datos pueden provenir de fuentes múltiples que contiene diversas imágenes y es necesario realizar el preprocesamiento y transformación en el dataset recopilado.

- **Preprocesamiento**

El preprocesamiento de imágenes consiste en modificar la representación original de una imagen digital mediante la transformación de tonalidades y colores. Los dos objetivos principales del preprocesamiento son: a) obtener una mejora en cuando a calidad de la imagen para el análisis de un especialista, y b) obtener información específica que ayudara en el procesamiento de cálculo (Pérez & Valente, 2018).

- **Transformación**

El objetivo de la transformación de una imagen es modificar el contenido de la misma, tales como eliminar el ruido presente en la imagen o extraer bordes. También se

realiza las transformaciones geométricas que consiste en modificar las coordenadas espaciales de la imagen para obtener diferentes resoluciones (Herrera et al., 2018).

### **Fase de entrenamiento**

El objetivo de esta fase es descubrir patrones insospechados, modelos que presentan algún tipo de relación, un conjunto de descubrimiento que contiene patrones secuenciales, clasificación y asociación. En esta exploración se hace uso de los algoritmos que toman de entrada un set de datos y en el proceso de análisis se descubren patrones de interés (Timarán Pereira et al., 2016).

En la fase de entrenamiento se ocupan modelos de aprendizaje automático como redes neuronales convolucionales para entrenar un set de imágenes y obtener resultados de clasificación, regresión o agrupación.

### **Fase de evaluación e interpretación**

En esta etapa se interpretan los datos, se visualiza los patrones extraídos donde existe patrones relevantes y redundantes, los resultados son traducidos para que el usuario pueda interpretarlos fácilmente. Este proceso de descubrimiento de conocimientos puede ser incorporado en conjunto con otro sistema para resolver problemas potenciales, que es de suma importancia el análisis profundo de información (Timarán Pereira et al., 2016).

## **1.4. Aplicativo**

### **1.4.1. Detalles del sistema**

El sistema propuesto para su desarrollo requiere de imágenes de rayos x, específicamente de la columna vertebral, es muy importante que la columna vertebral este completa a partir de su vértebra iniciar hasta su vértebra final. Las imágenes deben tener una calidad aceptable para su correcto etiquetado, esto favorecerá a un diagnóstico más aproximado.

A continuación, se especifica las fases que conforma el desarrollo del aplicativo, cada fase se describirá a profundidad en el Capítulo 2.



Fig. 21. Fases del desarrollo del aplicativo

En el aplicativo web el usuario que utilizará la aplicación será el especialista en escoliosis, el usuario podrá insertar nuevos pacientes y a su vez modificarlos. Por cada paciente cuenta con la opción de procesar imágenes radiografías, en esta etapa solo se necesita subir una imagen de rayos x de la columna vertebral y esperar el resultado final sobre el cálculo automático del ángulo de curvatura.

#### 1.4.2. Lenguaje de programación

El lenguaje a utilizar en el desarrollo del proyecto es Python. Este poderoso lenguaje es utilizado en distintas áreas de la ciencia como la inteligencia artificial, big data, Deep learning, web, entre otros. Es un lenguaje versátil multiplataforma y multiparadigma, destacado por su código limpio y legible, apropiado para manejar grandes volúmenes de datos. Python cuenta con una gran biblioteca de recursos a nivel científico y una comunidad inmensa que facilita la curva de aprendizaje y la solución de problemas más comunes.

#### Características del lenguaje Python

- **Lenguaje interpretado:** Cuando se ejecuta Python, el código fuente no es compilado a código máquina, mediante un intérprete ejecuta las instrucciones directamente. Es decir, el código fuente no realiza el proceso de compilación, el intérprete se encarga de ejecutar las instrucciones en tiempo real.
- **Multiplataforma:** Python puede ser ejecutando en distintos sistemas operativos, como Windows, Mac Os, Unix y Linux.

- **Multiparadigma:** Python soporta varios paradigmas de programación, como orientación a objetos, estructurada, programación imperativa y programación estructural. Su fuerte y el más usado es la programación orientada a objetos.
- **Tipado dinámico:** Python facilita la declaración de variables, no es necesario establecer el tipo de dato (int, double, char, string, float, etc) y su valor puede cambiar a otro tipo de dato en cualquier momento.

En el desarrollo del proyecto se utilizarán las siguientes librerías de Python.

- **Sys:** El módulo sys es el intérprete de Python, contiene muchos métodos y variables de funciones, entre ellos podemos interactuar con el sistema estando fuera del programa.
- **Os:** El módulo os permite interactuar con archivos y la estructura de directorios (leer y escribir) adaptando sus funcionalidades a los distintos sistemas operativos.
- **Math:** El módulo cuenta con funciones de operaciones complejas como fórmulas matemáticas, logaritmos, operaciones trigonométricas, entre otros. Su desarrollo está definido en el estándar C.
- **Numpy:** Es una librería que está especializada en soportar grandes cantidades de estructura de datos en formato matriz, permitiendo realizar cálculos numéricos, análisis de datos, etc. Realiza cálculos muy eficientes y es muy usado en análisis de datos científicos.
- **Tensorflow:** Es una plataforma de código abierto para construir y entrenar redes neuronales mediante el aprendizaje automático, es muy fácil de usar y crear diferentes arquitecturas de redes neuronales. Esta biblioteca permite entrenar y ejecutar redes neuronales profundas, utilizado para la clasificación y reconocimiento de imágenes, también es muy útil para entrenar redes neuronales recurrentes. La ejecución puede ser implementado mediante la CPU o GPU.  
Tensorflow crea estructuras de datos que se mueven mediante una serie de nodos de gráficos de procesamiento, estos nodos representan una operación matemática, cada conexión entre nodos o aristas es un tensor, es decir una matriz de datos multidimensionales (Buhigas, 2018).
- **Keras:** Es una biblioteca de código abierto creada en Python, utilizada para la creación de redes neuronales y permite hacer uso de varios frameworks de aprendizaje automático (IONOS, 2020).
- **Matplotlib:** Es una librería multiplataforma, permite realizar gráficas en 2D con facilidad y de alta calidad. Es utilizada en proyectos de Data Science obteniendo resultados atractivos e interactivos.

- **Flask:** Es un micro Framework desarrollado en Python, permite la creación de páginas web bajo el patrón MVC, la creación de APIs, entre otros. Es un buen punto de entrada por su fácil estructura en el desarrollo.
- **Skimage:** Es una biblioteca para procesamiento de imágenes, el uso de sus funciones es demasiado simple, perfecto para empezar en el mundo de visión por computadora.

## Anaconda

Anaconda es una distribución para Python, cuenta con un sistema de gestión de entornos, permite crear entornos con diferentes versiones de Python e instalar librerías sin modificar el entorno base del sistema. Anaconda es multiplataforma, se ejecuta en Linux, Windows y macOS.

### 1.4.3. Servidor Web

Los servicios web son ordenadores muy potentes que están en constante escucha de peticiones que realizan los usuarios, mediante dispositivos móviles, ordenadores, impresoras, entre otros. Estos ordenadores o servidores almacenan grandes volúmenes de información a diario. Los datos son transmitidos a los usuarios a través del protocolo HTTP (Hypertext Transfer Protocol) o HTTPS, estos datos son procesados por el navegador para otorgar una vista al usuario.

Para el desarrollo del proyecto se utilizó el lenguaje de programación JavaScript en el entorno de ejecución Nodejs. Este lenguaje tiene un gran desempeño con múltiples llamadas de funciones y permite tener una escalabilidad muy alta.

### 1.4.4. Base de datos

Una base de datos es una colección de información ordenada mediante una estructura que facilita el acceso a los datos de forma eficiente. Los datos son guardados en tablas y cada fila es un registro, las columnas son las cabeceras que da información sobre qué tipo de dato puede contener dicha columna. Una base de datos puede contener más de una tabla permitiendo tener un conjunto de información relacionada. Con respecto a los datos, existen reglas que permiten tener la información limpia, libre de datos nulos, datos repetidos, entre otros. La seguridad es muy importante en la base de datos, por ello existe reglas que otorgan permisos a usuarios específicos (Pulido Romero et al., 2019).

## Lenguaje SQL

- **Lenguaje definición de datos:** El lenguaje de definición de datos por sus siglas DDL (data definition language) permite gestionar la base de datos llevando a cabo las tareas de creación de la estructura de la base de datos. A continuación, se describe

los tipos de objetos (CREATE, DROP, ALTER, TRUNCATE) (Pulido Romero et al., 2019).

Tabla 2. Lenguaje DDL

Comando	Descripción
CREATE	Permite crear base de datos, tablas, campos, entre otros.
DROP	Eliminar tablas o índices.
ALTER	Permite cambiar la estructura de una tabla.
TRUNCATE	Elimina los datos de la tabla sin cambiar su estructura.

- Lenguaje de manipulación de datos:** Denominado como DML (data manipulation language) permite realizar tareas de consulta, eliminación, actualización, inserción. El lenguaje es proporcionado por los sistemas de gestores de base de datos facilitando la manipulación y recuperación de los datos mediante la estructura del lenguaje DML. A continuación, se describe los tipos de objetos (INSERT, DELETE, SELECT, UPDATE) (Pulido Romero et al., 2019).

Tabla 3. Lenguaje DML

Comando	Descripción
INSERT	Permite insertar registros en una tabla.
DELETE	Eliminar registros de una tabla.
SELECT	Realiza consultas de una o varias tablas.
UPDATE	Actualiza los registros de una tabla.

## PostgreSQL

Postgresql es una base de datos relacional, un sistema de código abierto y gratuito, adecuado para integrar proyectos avanzados y de muchos recursos. Es usada en todo el mundo por ser multiplataforma, fácil de implementar en diferentes lenguajes de programación, puede ser ejecutada en distintos sistemas operativos, tiene un gran rendimiento al manejar grandes volúmenes de datos.

Tomando en cuenta todas estas características, el proyecto propuesto implemento la base de datos Postgresql por su gran compatibilidad con el lenguaje de programación javascript.

### 1.4.5. Angular

Angular es un Framework especializado en el desarrollo de aplicaciones web permitiendo reutilizar código y contar con un alto rendimiento, ofrece un gran control en el desarrollo del Frontend. Cada parte de la funcionalidad de la interfaz está separado por

componentes para una manipulación específica y escalable, adecuada para manejar enormes volúmenes de data y renderizarlo en formularios. Angular también puede ser implementado para móvil nativo y escritorio nativo (Angular, 2022).

### **1.5. Trabajos relacionados**

Los trabajos relacionados basado en redes neuronales convolucionales se agrupo en tres categorías:

#### **En la Medicina**

##### **Técnicas de IA aplicadas a imágenes de rayos x para la detección de estructuras anatómicas de los pulmones y sus alteraciones.**

(Fos Guarinos, 2016) Aplicaron la arquitectura de CNN AlexNet para encontrar patologías en radiografías de tórax, para ello se enfocaron en detectar anomalías de cardiomegalia y efusión pleural. La investigación fue comparada con otro estudio previo, para su investigación utilizaron un dataset de 2242 imágenes superior al anterior estudio. Como resultado final demostraron que en la patología de anomalías obtuvieron un 89% de asientos superando con un 10% al anterior proyecto, un 91% de aciertos en patologías de cardiomegalia superando con un 7%, y por último en la patología de efusión pleural obtuvieron un 91% de aciertos superando al otro proyecto con 1%. En conclusión, la arquitectura CNN AlexNet supera un 7% al anterior estudio previo, pero no es equitativo en la gran cantidad de datos que tuvieron que utilizar para obtener dicho resultado.

##### **Reconocimiento de imágenes para detección temprana de alteraciones posturales**

(Romero Rodríguez, 2019) Su investigación se basa en detectar automáticamente la desviación de la columna vertebral en pacientes que se realizan rayos x de la zona lumbar, para ello utilizó la arquitectura Mask R-CNN con el modelo pre-entrenado COCO, con el objetivo de localizar la región de la columna vertebral y a partir de ahí calcular el ángulo de desviación. Los resultados obtenidos tras la detección de la zona de interés fueron con un 90% de predicción y lo más sorprendente es que contó con un dataset muy pequeño, con un tamaño de 100 imágenes radiográficas. En conclusión, Mask R-CNN con el modelo COCO su porcentaje de predicción es muy alta gracias a la gran cantidad de imágenes entrenadas previamente, por ello no se necesita de muchas épocas de entrenamiento cuando se agrega un nuevo objeto de detección.

## **Desarrollo de un clasificador basado en redes neuronales para la detección de escoliosis en imagen de rayos x de columna.**

(Martínez Ojeda, 2022) Realizo un trabajo que consiste en clasificar imágenes radiográficas con respecto a la patología de escoliosis, su clasificación se basó en escoliosis de tipo C y S, y columna normal. Contando con un dataset de 150 imágenes de rayos x, utilizó 75 imágenes para el entrenamiento y 75 imágenes para pruebas. Para su entrenamiento hizo uso de una red neuronal convolucional multicapa con dos capas de convolución. En su fase de pruebas obtuvo un porcentaje de aciertos del 94 %. Con este resultado permite que los fisioterapeutas y ortopedistas puedan tomar decisiones claves y con menor tiempo.

### **En la Gastronomía**

#### **Detección de frutas en árboles aplicando redes neuronales convolucionales basado en segmentación semántica**

(Torres Alonso, 2020) Utilizó la arquitectura de Faster RCNN, su investigación se enfocó en reconocer mandarinas con distintos colores (verde, naranja). Para ello hizo una comparativa entre un SSD y Faster RCNN, los resultados mostraron que Faster tiene una precisión media de 0.6 sobre 1.0 de SSD. Para reconocer objetos grandes supera SSD a Faster RCNN por su gran precisión, siendo más ligero y rápido. Pero Faster se desempeñó por la media con objetos pequeños.

#### **Generación de datos sintéticos con objetos de cocina para entrenar redes neuronales de convolución.**

(Benages-Pardo et al., 2019) Aplicaron la arquitectura de Mask RCNN para reconocer objetos de cocina, utilizando modelos pre-entrenados como los datos de COCO. Afirma que con una pequeña cantidad de datos se obtiene resultados aceptables. Para el entrenamiento utilizó más de 200 imágenes por objeto, obteniendo un resultado mayor del 86% de aciertos en reconocer objetos como botellas, sartenes, ollas y personas. Tomando en cuentas los resultados, la arquitectura Mask RCNN obtiene un alto porcentaje de aciertos a partir de un conjunto pequeño de datos entrenados.

### **En la Conducción**

#### **Detección y seguimiento de objetos para un vehículo en conducción.**

(García Fontán, 2019) Utilizó la arquitectura de Mask RCNN para el desarrollo de un sistema de detección y seguimiento de un vehículo en conducción, es un sistema de clasificación, donde define como **hard** cuando el vehículo se encuentra muy lejos y **easy**

cuando el vehículo se encuentra cerca de un obstáculo. En el resultado final obtuvieron las siguientes predicciones, en el apartado de **easy** un 0.86% de efectividad y un 0.60% de efectividad en el apartado de **hard**. El autor afirma que usar la arquitectura Mask RCNN al momento de realizar pruebas reales, el proceso de predicción de cada fotograma es demasiado lento, esto puede ocasionar actos indebidos del vehículo al tener un retraso en detectar un obstáculo antes de lo previsto.

A continuación, se presenta la matriz de conceptos con los proyectos relacionados.

		PROBLEMA/TÉCNICA				ALGORITMO				Resultados obtenidos
		Clasificación	Regresión	Agrupamiento	Asociación	CNN AlexNet	Mask R-CNN	Faster R-CNN	CNN	
Medicina	Belén Guarinos	X				X				La arquitectura fue entrenada con 2242 imágenes y obtuvieron un 7% de efectividad, superando a otros estudios previos que utilizaron un conjunto pequeño de datos.
	Claudia Verónica Romero		X				X			Mask RCNN con el modelo COCO su predicción es muy alta gracias a la gran cantidad de imágenes entrenadas previamente. Para entrenar un nuevo objeto no es necesario entrenar con miles de épocas, es suficiente con cientos.
	Martínez Gonzalo Enrique	X							X	El autor hizo uso de una red neuronal multicapa (dos capas convolucionales) para clasificar imágenes radiográficas entre escoliosis tipo C y S, y columna normal. Su dataset se conformó por 150 imágenes y obtuvo un porcentaje de
Gastronomía	Torres Alonso	X						X		Comparativa entre una red neuronal SSD y Faster RCNN, los resultados mostraron que Faster tiene una precisión media de 0.6 sobre 1.0 de SSD.
	Luis Benages Pardo	X					X			La arquitectura Mask RCNN obtiene una predicción mayor al 90% de efectividad a partir de un conjunto pequeño de datos entrenados.
Conducción	Javier García	X					X			El autor afirma que al usar la arquitectura Mask RCNN al momento de realizar pruebas reales, el proceso de predicción de cada fotograma es demasiado lento, esto puede ocasionar actos indebidos del vehículo al tener un retraso en detectar un obstáculo antes de lo previsto.

Fig. 22. Matriz de conceptos de proyectos relacionados

A partir de la matriz de conceptos, se obtiene una guía en la cual se identifica la mejor y adecuada arquitectura de red neuronal convolucional para ser implementada en este trabajo de investigación. En conclusiones se optó por la arquitectura Mask R-CNN, porque cuenta con varias tecnologías que se agrupan para la segmentación de instancias, adecuada para obtener la región de la columna vertebral. Esta CNN cuenta con un modelo pre-entrenado llamado COCO y tiene la facilidad de reconocer nuevas características de un nuevo objeto por entrenar, permitiendo reducir el número de imágenes que conforma el dataset.

El modelo Mask R-CNN en el proceso de entrenamiento cuenta con un porcentaje de predicción más del 90% y por medio de la GPU pude ejecutar 2 imágenes a la vez. El modelo es adecuado cuando no se cuenta con muchas imágenes para el entrenamiento.

## CAPÍTULO 2

### Desarrollo

La implementación del presente proyecto sobre el cálculo del ángulo de desviación de la columna vertebral en pacientes que padecen escoliosis, optimizara el tiempo del proceso que realizan los especialistas para calcular el ángulo de curvatura de forma manual. El aplicativo proporcionará un registro histórico de cada paciente para conocer la evolución y el avance de su condición. Este proyecto utilizara técnicas de visión artificial para la detección del contorno de la columna vertebral y se desarrollara un algoritmo especializado en calcular el ángulo de curvatura de forma automática utilizando el método Cobb.

#### 2.1. Visión general del proyecto

- **Suposiciones**

El desarrollo del proyecto dio inicio con la recopilación de imágenes radiográficas proporcionadas por varios centros de rayos x, para proceder con el entrenamiento de la red neuronal convolucional basada en regiones (Mask R-CNN). La información resultante dio paso para el desarrollo del algoritmo anterior mente mencionada.

En el proceso del cálculo de curvatura se contó con la ayuda de personas especializadas en la detección de escoliosis en pacientes que padecen esta condición.

- **Restricciones**

El tiempo del desarrollo del proyecto se encuentra estimado en 6 meses para su correcta distribución y ejecución, se organizó en un cronograma en la que se establece las actividades que se tiene que cumplir acorde al tiempo. Cada fase debe contar con el cumplimiento del desarrollo tomando en cuenta con la aprobación de los expertos.

#### 2.2. Entregables del proyecto

Los entregables del proyecto se detalla a continuación en la Tabla 4, se toma en cuenta las actividades por cumplir y el resultado que se obtiene al finalizar dicha actividad.

*Tabla 4. Entregables del Proyecto*

ENTREGABLE	DETALLE
Dataset	Fase de recolección de datos
Vista Minable	Fase de selección y etiquetado de imágenes
Entrenamiento	Fase de entrenamiento de la red neuronal convolucional
Interpretación de la información	Visualización de resultados
Algoritmo	Fase de desarrollo del algoritmo aplicando el método Cobb

## 2.3. Organización del proyecto

### 2.3.1. Participantes del Proyecto

A continuación, en la Tabla 5 se detalla los participantes directos en el proyecto.

Tabla 5. Directores de las Áreas Comprendidas

ROL	DEPENDENCIA	NOMBRE
Jefe de Proyecto	Carrera de Ingeniería en Sistemas Computacionales	PhD. Iván García
Especialista en calcular el ángulo de curvatura	Dueño del centro de rayos x	Lic. Wilson Cataña
Analista de Sistemas	Carrera de Ingeniería en Sistemas Computacionales	Sr. Carlos Cumbal

### 2.3.2. Roles y Responsabilidades

En la Tabla 6 se describe los roles asociados a cada uno de los participantes del proyecto con su respectiva responsabilidad.

Tabla 6. Roles y Responsabilidades del proyecto

ROL	DEPENDENCIA
Jefe de Proyecto	Es la persona responsable de gestionar el proyecto (a lo largo de su vida), tiene un conocimiento muy amplio de cada área que engloba el proyecto, se asegura que el proyecto esté bien planteado para así alcanzar los objetivos. Es la persona central en el tema de comunicación y finalmente realiza el seguimiento continuo para garantizar su éxito (Ocaña, 2013).
Especialista en calcular el ángulo de curvatura	El especialista en calcular la curvatura de la columna vertebral, cumple con la función de proporcionar información sobre que método utilizar y que pasos a seguir para conseguir un buen resultado en la medición del ángulo.
Analista del Sistemas	Encargado de analizar la información proporcionada por el especialista y transmitirlo mediante un software.

## 2.4. Gestión del Proyecto

### 2.4.1. Estimaciones

En la Tabla 7 se detalla la estimación del costo del talento humano, el método de estimación se basa por el costo por horas empleadas en la realización del proyecto.

Tabla 7. Talento Humano

DESCRIPCIÓN	N. DE HORAS	COSTO POR HORA (\$)	COSTO TOTAL (\$)
Horas de investigación del proyecto	160	20.00	3200.00
Horas de desarrollo del proyecto	220	20.00	4400.00
		<b>TOTAL</b>	7600.00

En la Tabla 8 se detalla el presupuesto estimado de los recursos que fueron necesarios para la elaboración del proyecto, entre ellos tenemos recursos de hardware, software, materiales de oficina e investigación.

Tabla 8. Recursos Materiales

DESCRIPCIÓN	COSTO REAL (\$)	COSTO ACTUAL (\$)
<b>Hardware</b>		
Computadora portátil	700.00	00.00
Impresora	210.00	00.00
<b>Software</b>		
Microsoft Word	00.00	00.00
Microsoft Excel	00.00	00.00
Google Colab	00.00	00.00
VGG Image Annotator	00.00	00.00
<b>Materiales de Oficina</b>		
Tinta de impresora	60.00	35.00
Hojas A4	04.50	04.50
Esferos	02.00	02.00
Internet	150.00	120.00
Flash Memory	20.00	00.00
<b>Investigación</b>		
Textos	15.00	00.00
Radiografías	20.00	20.00
<b>TOTAL</b>	1181.50	181.50

## 2.4.2. Plan del proyecto

El presente proyecto se utilizó la metodología KDD en el proceso de entrenamiento de la CNN para posteriormente continuar con el desarrollo del algoritmo, cada fase del proyecto refleja un resultado y el cumplimiento de los objetivos. En la Tabla 9 se detalla las fases por realizar con el tiempo de duración en horas.

Tabla 9. Fases del proyecto y su distribución en horas

FASE	DURACIÓN EN HORAS
<b>Metodología KDD</b>	
Fase de recolección de datos (imágenes radiográficas)	30
Fase de selección, preprocesamiento y transformación	15
Fase de entrenamiento de la red neuronal convolucional	40
Fase de evaluación e interpretación	10
<b>Desarrollo del algoritmo</b>	
Algoritmo utilizando el método Cobb	165
Desarrollo del Aplicativo	75
Muestreo	25
Análisis de Resultados	40
<b>TOTAL</b>	<b>400</b>

## 2.5. Fase de recolección de datos

Un dataset es un conjunto de datos estructurados, los datos pueden ser almacenados en una tabla donde se divide en filas y columnas, una fila puede ser un registro y las columnas las distintas variables de tipos de datos de cada elemento del registro. Ejemplos muy claros podemos encontrar un dataset en una hoja de Excel, una base de datos, archivos almacenados, un folder que comparte un mismo formato, entre otros. Los dataset son muy utilizados en las nuevas investigaciones tecnológicas como el Big Data, fundamentales para el procesamiento de datos. Para la creación del dataset se debe aplicar métodos que permiten depurar los datos evitando duplicaciones, datos nulos, datos incorrectos, etc.

Para el proceso de entrenamiento de la red neuronal convolucional se recolectó datos de imágenes radiográficas de la columna vertebral de pacientes de ambos géneros que presentan escoliosis. Las imágenes fueron proporcionadas por varios Centros de Rayos X, incluido el Centro de Rayos X del Cantón Cayambe en la que será implementado el aplicativo en el presente proyecto de investigación, también se obtuvieron imágenes de una base de datos de Kaggle. En el proceso de elaboración del dataset se tomó en cuenta que las imágenes de la columna vertebral estén completas y la resolución de las imágenes no sean de baja calidad.

El dataset está conformado por 300 imágenes de rayos x con columnas vertebrales completas, su distinción para el entrenamiento se aplicó de la siguiente manera.

Tabla 10. Dataset para entrenamiento

ID Objeto	Cantidad	Entrenamiento	Validación
columna_vertebral	300	225	75

La arquitectura Mask R-CNN cuenta con un modelo pre-entrenado llamado COCO, el modelo contiene miles de pesos que le facilitan reconocer nuevas características de un nuevo objeto, diferente a los diversos objetos existentes en el modelo, demostrando que con un porcentaje pequeño de datos se obtiene grandes resultados en la predicción.

## 2.6. Fase de selección, preprocesamiento y transformación

Al concluir con la recolección de imágenes radiográficas, se procedió a seleccionar y etiquetar.

### 2.6.1. Selección y Preprocesamiento

Los datos proporcionados por varios Centros de Rayos X se encontraban almacenados en un programa de visualización integrado en un DVD, se procedió a seleccionar solo imágenes de columna vertebral, ya que contaba con diversas partes radiográficas del esqueleto humano. En ocasiones se realizó limpieza en imágenes que tenían trazos sobrepuestos o descripciones de la información del paciente.

El mayor porcentaje de las imágenes obtenidas se tuvo que exportar de programas de visualización radiográfica como se muestra en la Fig. 23.

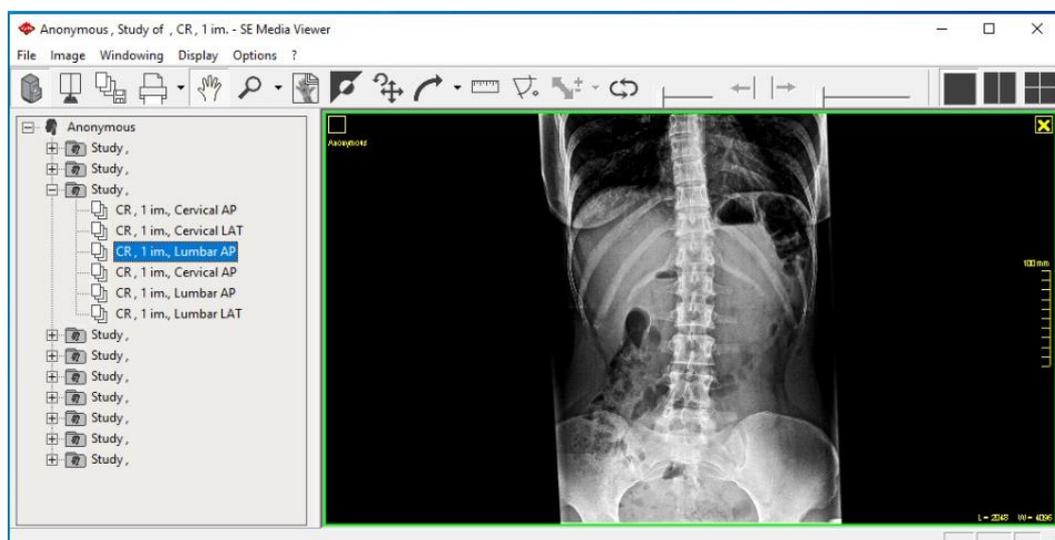


Fig. 23. Programa de visualización de radiografías

## 2.6.2. Transformación

El software VGG Image Annotator es un proyecto de código abierto desarrollado en Visual Geometry Group, permite definir regiones en una imagen y dar una descripción textual de la región etiquetada. Todos estos datos son útiles para el entrenamiento supervisado de algoritmos de aprendizaje (Dutta, 2018).

Este software permite etiquetar las imágenes con facilidad, existe diferentes tipos de etiquetado, entre ellos encontramos la forma de un cuadrado, círculo, polígono, entre otros. En la construcción del dataset etiquetado se utilizó trazos de polígonos para marcar el área del objeto y a continuación se estableció el nombre clave de su clasificación, en este caso corresponde al nombre “columna\_vertebral”, la herramienta permite etiquetar a más de un objeto dentro de la misma imagen.

Como resultado del dataset etiquetado, el software permite descargar en diferentes formatos la información procesada, en el desarrollo de este proyecto se descargó un archivo json que posteriormente se cargara en el modelo de entrenamiento. El archivo json tiene información guardada sobre el nombre de su clasificación, una lista de coordenadas con respecto a los puntos trazados en las áreas de cada objeto.

En la Fig. 24 se observa el etiquetado de las imágenes radiográficas de la columna vertebral en pacientes que padecen escoliosis.

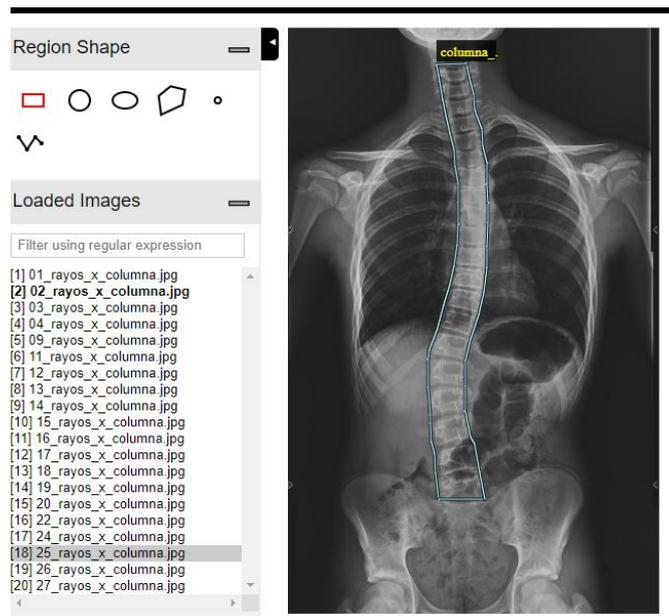


Fig. 24. Etiquetado de las imágenes

## 2.7. Fase de entrenamiento de la red neuronal convolucional

### 2.7.1. Archivos de configuración

Para obtener el código fuente de la arquitectura de Mask R-CNN se puede descargar en GitHub, para su instalación es adecuado contar con un entorno de desarrollo, en este caso se utilizará anaconda. Es muy importante que, en los requisitos de instalación las versiones sean compatibles con las distintas librerías instaladas.

Para la implementación del modelo Mask R-CNN se realizó las debidas configuraciones en el archivo de un modelo de entrenamiento, por ello se importará la arquitectura de Mask R-CNN y el directorio raíz del proyecto.

```
40
41 # Importar la ruta raíz del modelo Mask RCNN
42 sys.path.append(ROOT_DIR)
43
44 from mrcnn.config import Config
45 from mrcnn import model as modellib, utils
46
47 # Ruta del archivo que contine los pesos entrenados
48 COCO_WEIGHTS_PATH = os.path.join(ROOT_DIR, "mask_rcnn_coco.h5")
49
50 # Directorio para guardar registros
51 DEFAULT_LOGS_DIR = os.path.join(ROOT_DIR, "logs")
52
```

Fig. 25. Importación de la arquitectura Mask RCNN

En la configuración se integra un archivo llamado “mask\_rcnn\_coco.h5”, este archivo es un modelo pre-entrenado con un gran conjunto de datos para la detección, clasificación y segmentación de objetos, contiene todos los pesos que facilitara la detección de nuevos objetos etiquetados en el momento de su nuevo entrenamiento. También se especifica el directorio donde se guardará los nuevos modelos de entrenamiento, en este caso es la carpeta “logs”.

A continuación, se realizará la configuración para el entrenamiento de la red neuronal convolucional, para ello tenemos que especificar los siguientes parámetros:

- NAME: se especifica el nombre del tipo de clasificación.
- IMAGES\_PER\_GPU: se especifica el número de imágenes por procesamiento, se debe tomar en cuenta si en caso de colocar el número 2, la computadora debe contar con 12 GB de memoria RAM.
- NUM\_CLASSES: se especifica el número de clases incluyendo el fondo.
- STEPS\_PER\_EPOCH: el número de pasos de entrenamiento por cada época.

- DETECTION\_MIN\_CONFIDENCE: Omite la detección al detectar que cuenta con 90% de confianza.

```

58 class CustomConfig(Config):
59     """Configuración para entrenamiento sobre el conjunto de datos.
60     """
61     # Dar a la configuración un nombre reconocible
62     NAME = "columna_vertebral"
63
64     # Número de imágenes por proceso, Usar una GPU con memoria de 12GB
65     IMAGES_PER_GPU = 2
66
67     # Número de clases (incluidos el fondo)
68     NUM_CLASSES = 1 + 1 # fondo + nombre de clasificación
69
70     # Número de pasos de entrenamiento por época
71     STEPS_PER_EPOCH = 100
72
73     # Omitir detecciones con <90% de confianza
74     DETECTION_MIN_CONFIDENCE = 0.9
75

```

Fig. 26. Configuración para entrenamiento

Por último, se carga el dataset de entrenamiento y de validación. La carpeta dataset debe estar separado por dos carpetas llamadas train (entrenamiento) y val (validación), y cada una de ellas debe contener el archivo json con su respectiva información. También podemos configurar el número de épocas del entrenamiento, no es necesario entrenar todo el modelo en una sola ejecución, podemos realizarlo en diferentes tiempos de ejecución y así evitar el exceso uso de recursos del computador. En la Fig. 27 se observa la configuración anteriormente mencionada.

```

176 def train(model):
177     """Entrenar modelo."""
178     # dataset de entrenamiento.
179     dataset_train = CustomDataset()
180     dataset_train.load_custom(args.dataset, "train")
181     dataset_train.prepare()
182
183     # dataset de validación
184     dataset_val = CustomDataset()
185     dataset_val.load_custom(args.dataset, "val")
186     dataset_val.prepare()
187
188     print("Training network heads")
189     model.train(dataset_train, dataset_val,
190               learning_rate=config.LEARNING_RATE,
191               epochs=20,
192               layers='heads')
193

```

Fig. 27. Entrenar modelo

## 2.7.2. Entrenamiento

Es muy importante instalar los requisitos que solicita el modelo, principalmente las librerías de tensorflow y keras, se debe tomar en cuenta las versiones de cada librería para evitar conflictos de compatibilidad. Instalar estas herramientas en Windows puede provocar incompatibilidad y errores en el uso de las librerías, por ello en el desarrollo del proyecto se utilizó el sistema operativo Linux, las librerías fueron instaladas en un entorno de desarrollo que ofrece anaconda, con el fin de evitar modificaciones en el entorno base del sistema.

Al dar por concluido la configuración, se procede a ejecutar el fichero `columna_vertbral_train.py`, este programa cargará la arquitectura de la red neuronal convolucional y el modelo pre-entrenado permitiendo generar un nuevo modelo que reconocerá el objeto previamente especificado. En la Fig. 28 se puede visualizar el comando aplicado para la ejecución del entrenamiento.

```
(app_angulo_cobb) [roptu@fedora ~]$ python columna_vertbral_train.py train --dataset=/home/roptu/Documentos/Mask_RCNN-master/samples/columna_vertbral/dataset --weights=coco
```

Fig. 28. Comando para entrenar el modelo

El entrenamiento del modelo se dividió en diferentes partes para evitar el sobrecalentamiento del ordenador, por ello se especificó las épocas limitantes en cada ejecución. A continuación, en la Fig. 29 se visualiza el comando para reanudar el entrenamiento de la red neuronal convolucional.

```
(app_angulo_cobb) [roptu@fedora ~]$ python columna_vertbral_train.py train --dataset=/home/roptu/Documentos/Mask_RCNN-master/samples/columna_vertbral/dataset --weights=last
```

Fig. 29. Comando para reanudar el entrenamiento

Se utilizó una computadora con las siguientes características, un procesador Intel Core i7 de octava generación, RAM de 8 GB, un disco sólido, el entrenamiento de la red neuronal fue procesado por la CPU. Dependiendo de las características del ordenador el tiempo de entrenamiento varía, es muy importante que el equipo esté en un ambiente apropiado para su ejecución y es recomendable utilizar varios ventiladores o refrigerantes para evitar sobrecalentamiento en los componentes hardware y provocar el reinicio y/o apagado del ordenador, errores, pérdida de información y en casos extremos componentes inutilizables. Otra forma de entrenar la red neuronal es por GPU, para este caso se requiere una tarjeta gráfica que cuente como mínimo 6 GB de memoria RAM, su procesamiento es mucho más rápido que la CPU por su gran cantidad de núcleos al realizar cálculos matemáticos.

Con el ordenador especificado, el tiempo que tomo por cada imagen al ser procesada es entre 30-45 segundos. Existe soluciones alternativas cuando no se cuenta con un ordenador de altos recursos para el entrenamiento, se puede utilizar Google Colab que permite ejecutar código arbitrario de Python en el navegador.

También se utilizó un cuaderno de Google Colab para continuar con el entrenamiento de la red neuronal convolucional, fue necesario habilitar el procesamiento por GPU y así obtener un mejor rendimiento. En la Fig. 30 se visualiza la habilitación de la GPU en Google Colab.

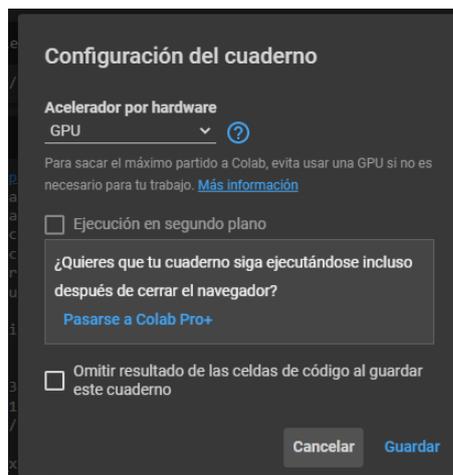


Fig. 30. Configuración de uso de GPU

De este modo se reanudó el entrenamiento de la CNN en Google Colab, el proceso fue mucho más rápido al ocupar la GPU, para almacenar el modelo de entrenamiento se vinculó con una cuenta de Google Drive que permite hasta 15 GB gratis de almacenamiento. En la Fig. 31 se aprecia la reanudación del entrenamiento.

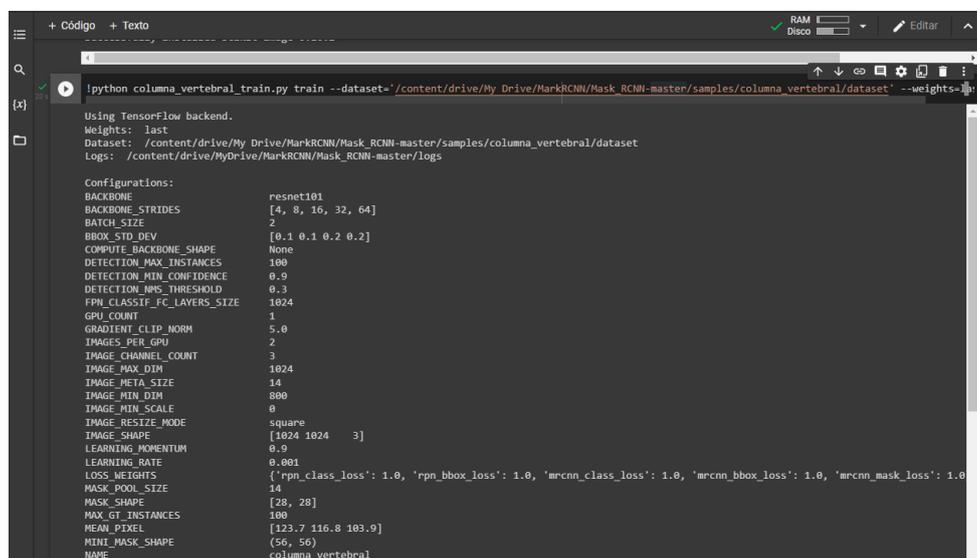


Fig. 31. Entrenamiento en Google Colab

## 2.8. Fase de evaluación e interpretación

El modelo Mask R-CNN está implementado en Python 3, utilizando las librerías de Keras y TensorFlow. Para detectar objetos en una imagen, el modelo genera cuadros delimitadores que permiten reconocer rasgos o trazos de un objeto etiquetado y a partir de ahí generar una máscara de segmentación para cada instancia de los objetos detectados en la imagen. Mask R-CNN utiliza una red neuronal profunda ResNet101 y Feature Pyramid Network (FPN), esta permite reconocer objetos en varias escalas (Abdulla, 2017).

A continuación, se describe las etapas para obtener la máscara del objeto en una imagen.

### 2.8.1. Predicciones

La red de propuestas de región Region Proposal Network (RPN), es una red neuronal muy ligera, mediante un clasificador binario que cubre toda la imagen con cuadros delimitadores denominados anclajes permite obtener las predicciones de objeto/no objeto. En la Fig. 32 se observa los cuadros de anclas en toda la imagen y una muestra de predicciones aleatoria.

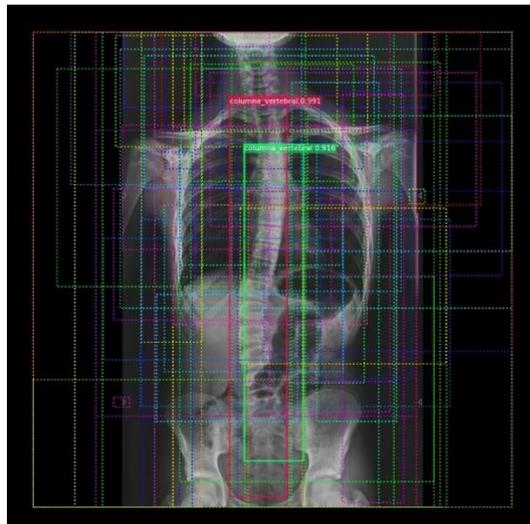
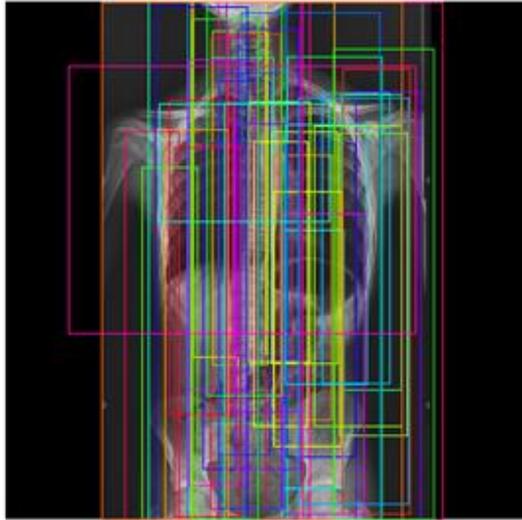


Fig. 32. Anclajes y sus predicciones

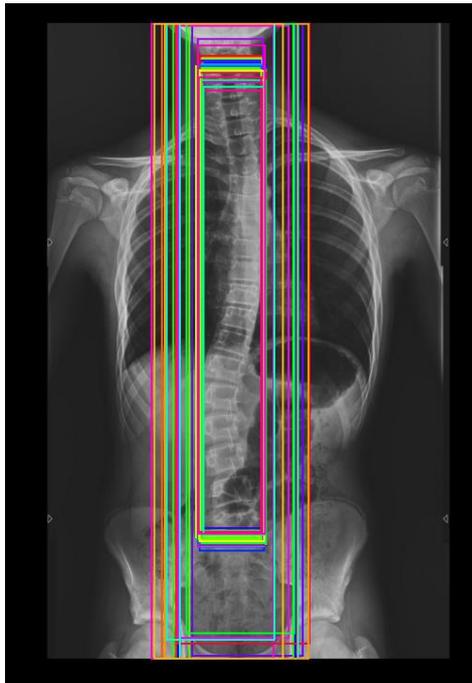
En la Fig. 33 se visualiza un refinamiento de anclas y un recorte a los límites de la imagen.



*Fig. 33. Refinamiento de anclas*

Para obtener las anclas positivas, la red neuronal analiza las cuadrículas en diferentes escalas para calcular el IoU (Intersection over Union). IoU es un estándar de medición que indica el grado de similitud entre los datos del objeto etiquetado y un anclaje. Los anclajes positivos son aquellos que tiene el  $\text{IoU} \geq 0.7$ , los anclajes negativos son los que contienen el  $\text{IoU} < 0.3$  denominados como no objeto y los anclajes entre  $\text{IoU} \geq 0.3$  y  $\text{IoU} < 0.7$  son considerados neutros y son excluidos en el entrenamiento.

En la Fig. 34 se observa los anclajes positivos con el  $\text{IoU} \geq 0.7$



*Fig. 34. Anclajes positivos*

## 2.8.2. Clasificación de propuestas

Por lo general los anclajes positivos no cubren la totalidad del objeto, existen varios anclajes que reconocen diferentes escalas del objeto etiquetado. En la Fig. 35 se visualiza los anclajes positivos con su predicción alrededor del objeto completo.

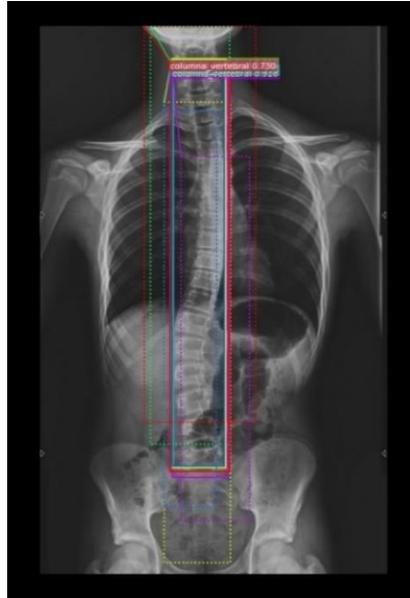


Fig. 35. Anclajes positivos y sus predicciones

Entonces, para obtener delimitado el objeto con un solo recuadro, el modelo RPN retrocede un refinamiento de los anclajes positivos para cambiarlo, después ajustar su tamaño y ubicación a los límites reconocidos del objeto etiquetado. Dando como resultado un recuadro que recubre todo el objeto, tal como se muestra en la Fig. 36.



Fig. 36. Predicción del objeto

### 2.8.3. Generación de mascarar

En esta etapa se utiliza los cuadros delimitadores refinados y su id de clase de la detección anterior, para ejecutar el encabezado de la máscara permitiendo generar la segmentación para cada instancia reconocida. En la Fig. 37 se aprecia el resultado de la máscara.



*Fig. 37. Mascara del objeto*

Finalmente, el modelo devuelve como resultado el objeto detectado, identificado mediante un cuadro delimitador en su ubicación y la máscara pixelada de la región del objeto, también se visualiza la puntuación de la predicción. Esta información se observa en la Fig. 38.



*Fig. 38. Resultado final de la segmentación de instancias*

## 2.9. Algoritmo utilizando el método Cobb

### 2.9.1. Regresión Polinomial

La regresión polinomial es un proceso estadístico para evaluar la relación entre variables independientes X y variables dependientes Y, su función resultante es modelada a partir de un polinomio de grado n en X. La regresión polinomial permite ajustar una curvatura a un modelo no lineal mediante el valor de X y la variación de la variable dependiente Y en torno a la función de regresión descrita como un espacio de probabilidades. El análisis de regresión es utilizado en el campo de aprendizaje automático, permite comprender qué relación tiene las variables dependientes a partir de las independientes, un ejemplo muy específico de uso, es determinar la progresión de epidemias que causaron muertes a causa de una enfermedad (Aragón Cervantes, 2014).

La función obtenida al aplicar regresión polinomial a un modelo de datos es la siguiente.

$$y = a_0 + a_1x + a_2x^2 + a_3x^3 + \dots + a_nx^n$$

En la Fig. 39 se observa varios ejemplos comparativos en ajustar los datos en diferentes modelos, entre ellos tenemos un modelo lineal puro, un modelo cuadrático y por último un modelo polinómico de 300 grados.

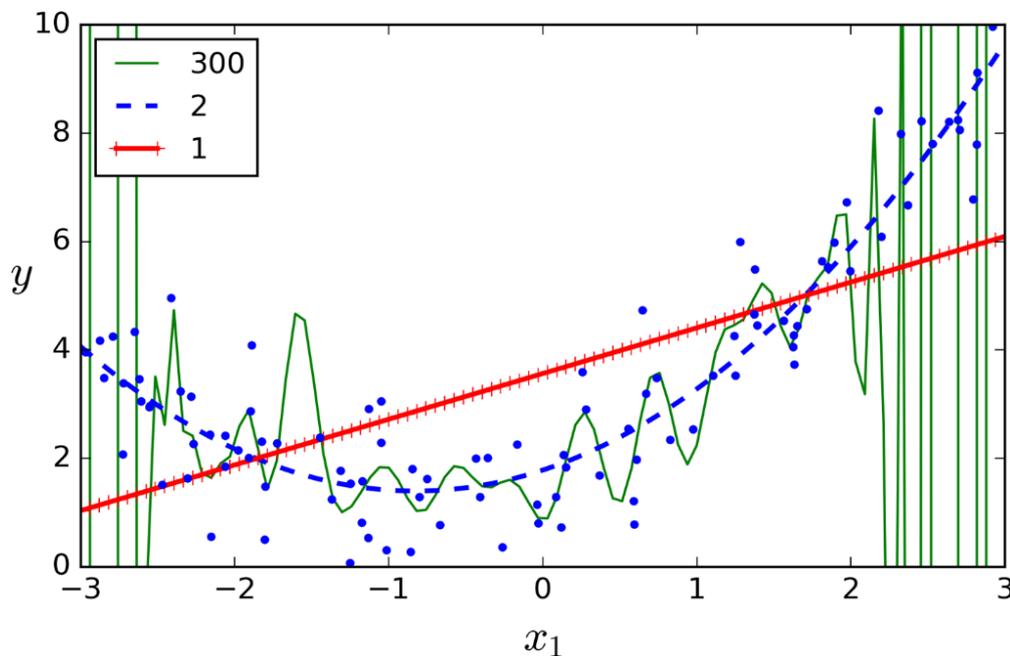


Fig. 39. Curvas de aprendizaje  
Fuente: (Géron, 2017)

Es importante enfatizar que, a medida que aumentamos el grado polinomial la curvatura tiende a justarse drásticamente al modelo, provocando la dificultad de detectar el sentido de la curva y la pérdida del rumbo de las variables dependientes resultantes al realizar un estudio probabilístico.

También cabe recalcar, los polinomios extremadamente elevados pueden producir errores al momento de resolver el sistema de ecuaciones, porque al tener números muy altos y a su vez un alto grado polinomial el método no podría resolverlo.

### 2.9.2. Creación del modelo de la columna vertebral

El objetivo consiste en encontrar una función que mejor se ajuste a los datos, es decir una función cuya representación gráfica sea una curva que se acerque de la mejor manera a los datos dados.

Para construir el modelo de la columna vertebral, se necesita obtener la zona de interés (columna vertebral). La red neuronal convolucional previamente entrenada y dado como input una imagen radiográfica, retorna una matriz con el mismo tamaño de fila y columna de la imagen de origen, esta matriz nueva se encuentra con valores entre 0 y 1, el valor 0 corresponde al fondo de la imagen y el valor 1 la máscara del objeto detectado. En este resultado de detección también cuenta con la información sobre a qué clase de id etiquetada pertenece y la puntuación del porcentaje de predicción. En la Fig. 40 se muestra el resultado de la predicción.



Fig. 40. Resultado de predicción del modelo Mask R-CNN

Una vez obtenido la zona de la columna vertebral, se procedió a obtener las coordenadas X y Y de la parte central de la columna, con la finalidad de crear un modelo acorde a las curvaturas del objeto reconocido. A continuación, en la Fig. 41 se muestra el

resultado de la obtención del modelo de la columna vertebral a partir de una lista de coordenadas.

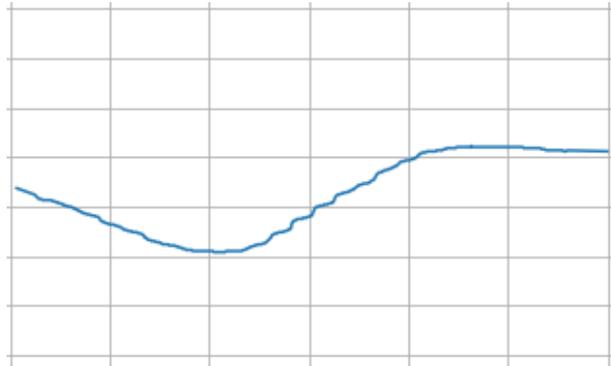


Fig. 41. Modelo columna vertebral mediante coordenadas

Como se puede observar en la anterior figura, el modelo no posee curvas uniformes, las curvaturas son con mucho ruido y difícilmente se puede trazar restas tangentes, la solución es aplicar regresión polinomial para ajustar la curva acorde a los datos.

Para obtener el grado del polinomio que mejor se ajuste a las curvas se realizó varias pruebas, el cual se optó por el grado 12, este grado fue el más exacto en ajustar al modelo de la columna vertebral de forma natural. Aplicando grados inferiores el resultado fue inapropiado, provocando pérdida de información en resaltar las curvaturas de la columna. Pero al aplicar grados superiores los datos resultantes perdían el sentido de la curva.

En la Fig. 42 se aprecia el resultado del modelo de la columna vertebral aplicando 12 grados previamente dado como selección final.

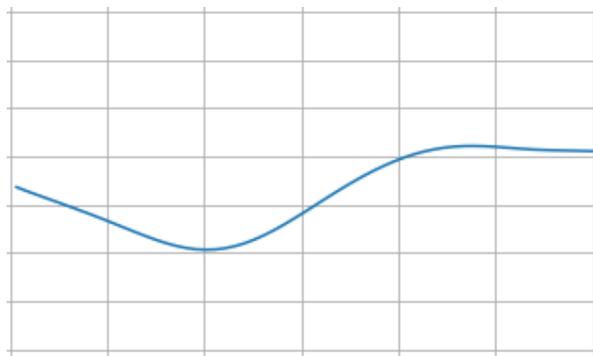


Fig. 42. Regresión polinomial con 12 grados

Para dar una demostración aplicando grados inferiores se obtiene el siguiente resultado en la Fig. 43.



Fig. 43. Regresión polinomial con 5 grados

Por último, una demostración aplicando grados muy altos, se visualiza el resultado en la Fig. 44.

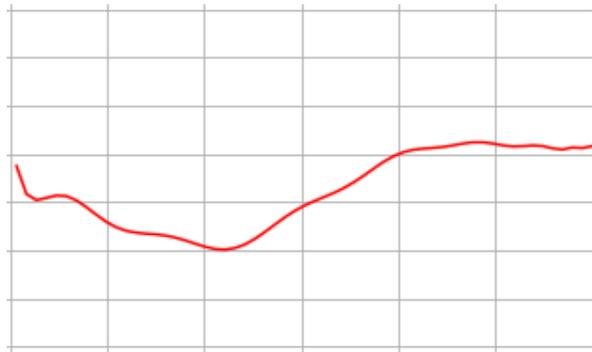


Fig. 44. Regresión polinomial con 35 grados

### 2.9.3. Cálculo del ángulo de desviación de la columna vertebral

Para realizar el cálculo del ángulo de la columna vertebral de forma automática a partir del modelo obtenido de la regresión polinomial, se aplicó las siguientes operaciones matemáticas.

#### Derivada

La derivada es una función matemática que significa que tan rápido se produjo una variación de cantidad en un determinado punto de una función. Permite resolver problemas como la velocidad y aceleración de un vehículo, el cálculo de áreas, como también el cálculo de la recta tangente de una curva en un punto (Casteleiro Villalba, 2013).

#### Ecuación de la recta tangente y recta normal

Una recta es tangente a una función cuando pasa por el punto seleccionado y tiene por pendiente el cálculo de la derivada de esa función aplicada al mismo punto. La recta tangente en un determinado punto de una curva, el punto tiene la misma pendiente que la curva (Fernández L, 2016). La expresión de la recta tangente es:

$$y - f(a) = f'(a) \cdot (x - a)$$

La resta normal es una recta perpendicular a la recta tangente a partir de un punto. La expresión de la recta normal es:

$$y - f(a) = -\frac{1}{f'(a)} \cdot (x - a)$$

A continuación, en la Fig. 45 se observa de manera gráfica el trazo de la recta tangente y normal en un plano cartesiano.

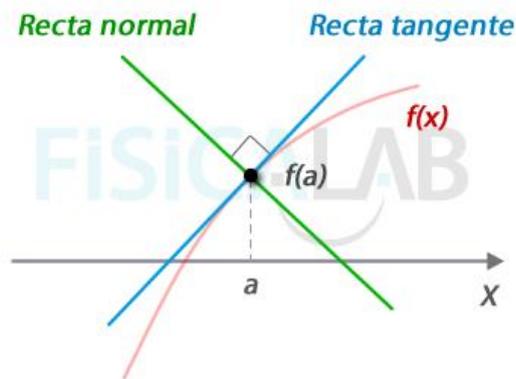
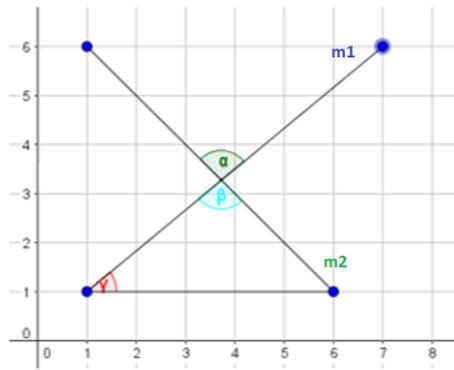


Fig. 45. Recta tangente y recta normal  
Fuente: (Fernández L, 2016)

### Ángulo entre dos rectas

Al tener dos rectas que se cortan entre sí en un plano cartesiano se puede calcular el ángulo, por lo general se generan dos ángulos, un agudo y otro obtuso, como también pueden ser perpendiculares al tener  $90^\circ$  en sus dos ángulos. Se puede calcular el ángulo menor de dos formas: mediante sus vectores directores o por sus pendientes. La manera más rápida y fácil es a partir de sus pendientes, al contar con los coeficientes  $m_1$  y  $m_2$  se aplica la fórmula y como resultado se obtiene un ángulo siendo  $0 < \alpha \leq \pi/2$  (Serra Requena, 2021).

En la Fig. 46 se visualiza la fórmula para obtener el ángulo entre dos rectas a partir de sus pendientes.



$$\tan \alpha = \frac{m_2 - m_1}{1 + m_1 m_2}$$

Fig. 46. Ángulo entre dos rectas

### Desarrollo del algoritmo

El algoritmo en su totalidad fue desarrollado en Python, ocupando las librerías anteriormente mencionadas. Tomando como guía y prioridad el método del Ángulo de Cobb. En su fase de desarrollo del algoritmo se realizó una gran cantidad de pruebas con imágenes radiográficas de diferentes casos de escoliosis, dando como resultado en imágenes de rayos x la obtención de dos ángulos. Para dar mayor detalle sobre el proceso matemático elaborado, se describe las siguientes etapas principales:

- 1) **Modelo de la columna vertebral:** Aplicando regresión polinomial en una lista de coordenadas de un modelo con mucho ruido en sus datos, dio como resultado una función que mejor se ajusta a los datos, en el cual se aplicó derivadas a dicha función para obtener la pendiente en cualquier punto, útil para trazar rectas tangentes.
- 2) **Localizar el inicio y final de la curvatura:** Para obtener los puntos donde inicia y termina la curvatura, se trazó una recta vertical denominada línea plomada tomando como base la parte final de la columna. A partir de ahí se realizó un barrido en cada punto de la ecuación de la columna vertebral para encontrar un cruce con la línea plomada. Por lo general se identificó dos cruces, dando como resultado el punto de inicio y el punto final de la curvatura.
- 3) **Trazo de la recta tangente y normal:** Teniendo como entrada los puntos que identifican la curvatura, se procedió a calcular la recta tangente y normal en cada punto, consiguiendo cuatro rectas que en un determinado espacio se intersecan entre sí. Es importante enfatizar que en los diversos tipos de escoliosis se necesitó ajustar parámetros para obtener un mejor resultado en trazar las rectas tangentes.
- 4) **Obtención del ángulo:** A partir de los trazos de las rectas tangentes localizados en el inicio y final de la curvatura de la columna, se calculó el ángulo con facilidad, con la ayuda de sus pendientes se aplicó la fórmula de ángulo entre dos rectas explicado anteriormente.

En la Fig. 47 se muestra el resultado al combinar todas las etapas desarrolladas en el lenguaje Python.

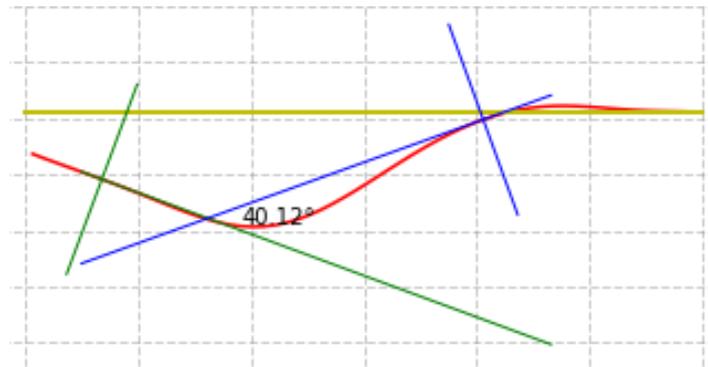


Fig. 47. Cálculo automático del ángulo de curvatura

#### 2.9.4. Resultados del algoritmo desarrollado.

Como resultados en procesar y aplicar el algoritmo en distintas patologías de imágenes radiográficas se obtuvo diferentes tipos de escoliosis, entre ellos encontramos curvatura de tipo normal y escoliosis de tipo leve, moderada y grave. A continuación, se presentan ejemplos.

En la Fig. 48 se visualiza una curvatura de tipo normal, el paciente es de tercera edad.

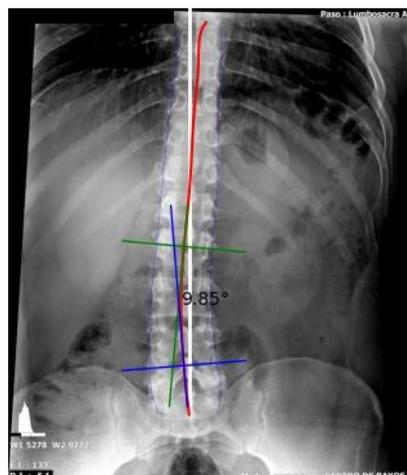


Fig. 48. Curvatura de tipo normal con 9.85 grados

En la siguiente Fig. 49 se visualiza una escoliosis de tipo leve.

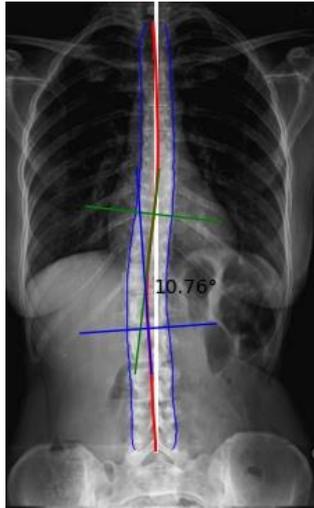


Fig. 49. Escoliosis de tipo leve con 10.76 grados

En la Fig. 50 presenta una escoliosis de tipo grave.

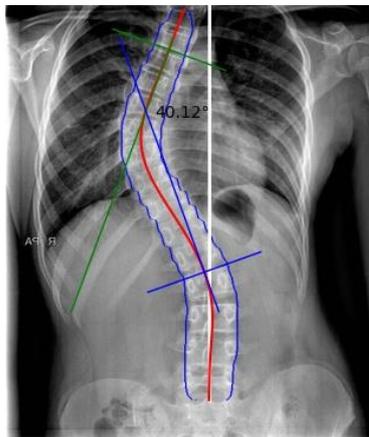


Fig. 50. Escoliosis de tipo grave con 40.12 grados

Por último. En la Fig. 51 se observa una radiografía que tiene como resultado dos ángulos, escoliosis de tipo leve y moderada respectivamente.

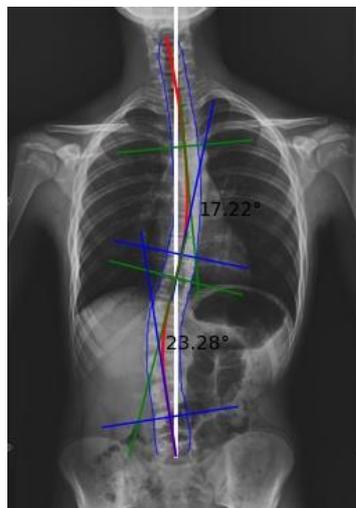


Fig. 51. Escoliosis de tipo leve y moderada con 17.22 y 23.28 grados respectivamente

## 2.10. Desarrollo del Aplicativo

Es necesario contar con una aplicación, en la cual el usuario puede interactuar de forma amigable con este proyecto de investigación. Por ello se desarrolló una aplicación web, esta aplicación cuenta con un control de pacientes y sus respectivos registros históricos correspondiente a su patología. En la Fig. 52 se visualiza la arquitectura del aplicativo implementado.

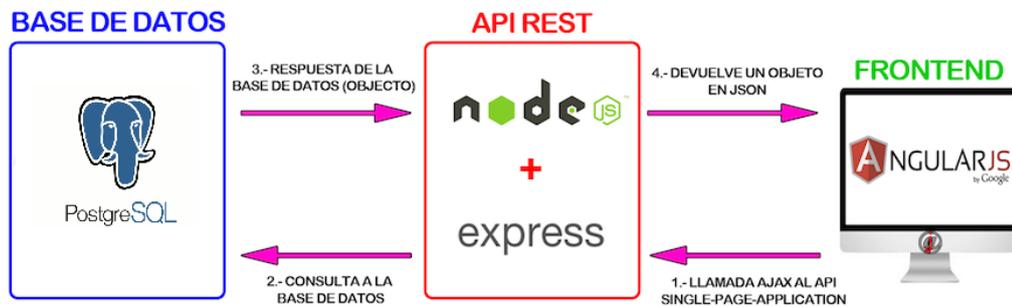


Fig. 52. Arquitectura aplicativo

El aplicativo desarrollado solo se ejecutará de forma local, la máquina que usará la aplicación contará con todas las librerías instaladas, incluido el gestor de base de datos.

### 2.10.1. Diseño de la base de datos

Para la creación del modelo de la base de datos, fue necesario plantear los requerimientos del sistema, se crearon varias tablas para el registro de pacientes y el procesamiento de imágenes radiográficas. El software utilizado para el modelado fue SqlDBM Enterprise, permite desarrollar bases de datos en línea, extraer código SQL a partir del modelo y es muy fácil de usar e interactivo. En la Fig. 53 se visualiza el esquema físico de la base de datos.

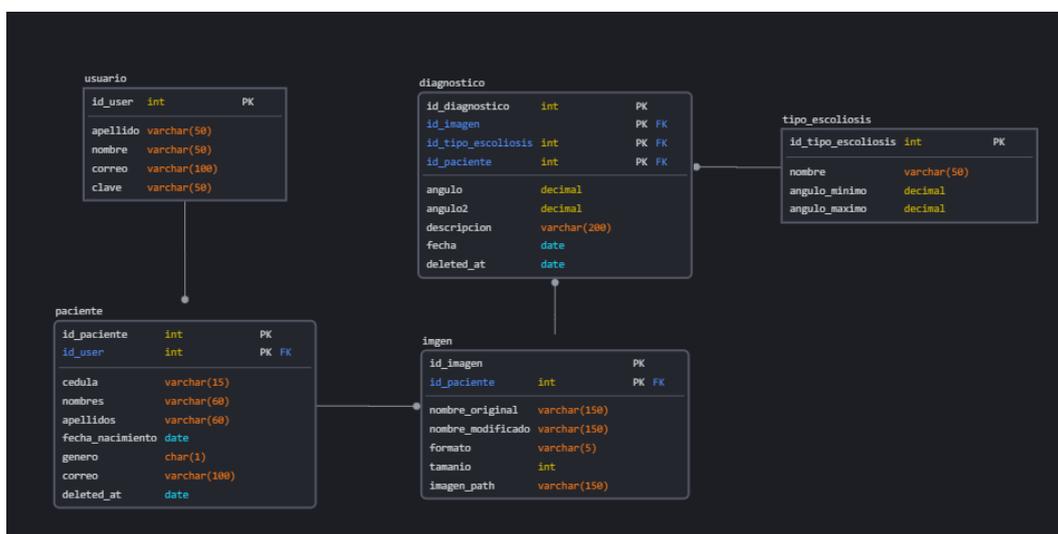


Fig. 53. Diseño base de datos

La base de datos solo será administrada por el usuario, es decir por el especialista en detectar patologías de escoliosis: El usuario tendrá privilegios de lectura y escritura en las tablas principales. Los registros que el usuario elimine, solo se ocultaran evitando perdidas de información mal intencionadas.

### **2.10.2. Desarrollo Backend**

Para el desarrollo de la API se utilizó el lenguaje de programación JavaScript en el entorno de ejecución Nodejs, este entorno permite ejecutar código js a lado del servidor. La librería pg-promise permitió realizar la conexión con la base de datos Postgresql de forma sincrónica. Y para el manejo de peticiones HTTP (POST, GET, UPDATE y PUT) se utilizó el framework Express, esta herramienta es muy potente para atender al mismo tiempo múltiples peticiones solicitadas por diversos tipos de consumidores. A través del método POST la API recibe el fichero de la imagen y lo almacena en el servidor.

Por último, en el Backend también incluye la aplicación del cálculo del Ángulo de Cobb desarrollado en el lenguaje Python, mediante la librería Flask se creó una API para la comunicación con el servidor realizado en Nodejs, su comunicación es mediante objetos json. La API en Python requiere la ruta del fichero de la imagen, para dar comienzo con el procesamiento de la imagen radiográfica y a su vez calcular el ángulo de curvatura, finalmente retorna a la API de Nodejs una nueva ruta con la ubicación del fichero de la imagen procesada.

### **2.10.3. Desarrollo Frontend**

Como al principio se mencionó. Angular se implementará en el desarrollo web, además de ofrecer una estructura ordenada al momento de codificar cada componente del aplicativo y sobre todo la reutilización de código, tiene soporte por Google y posee una gran comunidad que han descrito las soluciones de problemas más comunes. Cuando el proyecto desarrollado se pretende pasar a producción, se puede lograr mediante una línea de comando, la cual crea una carpeta con los archivos html, css y js de forma cruda. Solo falta cargar los archivos en cualquier servidor web o alojarlo en un servicio en la nube para su funcionamiento.

### **2.10.4. Presentación de resultados en el Aplicativo Web**

Al dar por concluido el análisis de imágenes radiográficas de la columna vertebral en pacientes que padecen escoliosis, se presenta los resultados en el aplicativo web. Este aplicativo permite ingresar, consultar, editar y eliminar pacientes. A su vez, en el entorno de

procesamiento de imágenes realiza las mismas acciones. A continuación, se presenta las visualizaciones gráficas del funcionamiento del aplicativo web:

- a) **Inicio de sesión:** Para mantener protegido el uso del aplicativo, se realizó un login que cumple con la función de dar acceso a usuarios autorizados, en el cual se solicita el correo y la clave. En la Fig. 54 se puede visualizar el componente login.

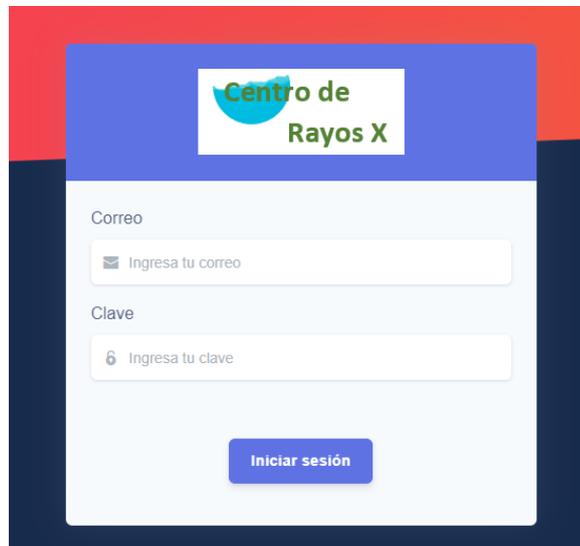


Fig. 54. Inicio de sesión

- b) **Ingresar pacientes:** Los datos del paciente solicitados son: cédula, nombres, apellidos, teléfono, fecha nacimiento, género y el correo electrónico, como se muestra en la Fig. 55.

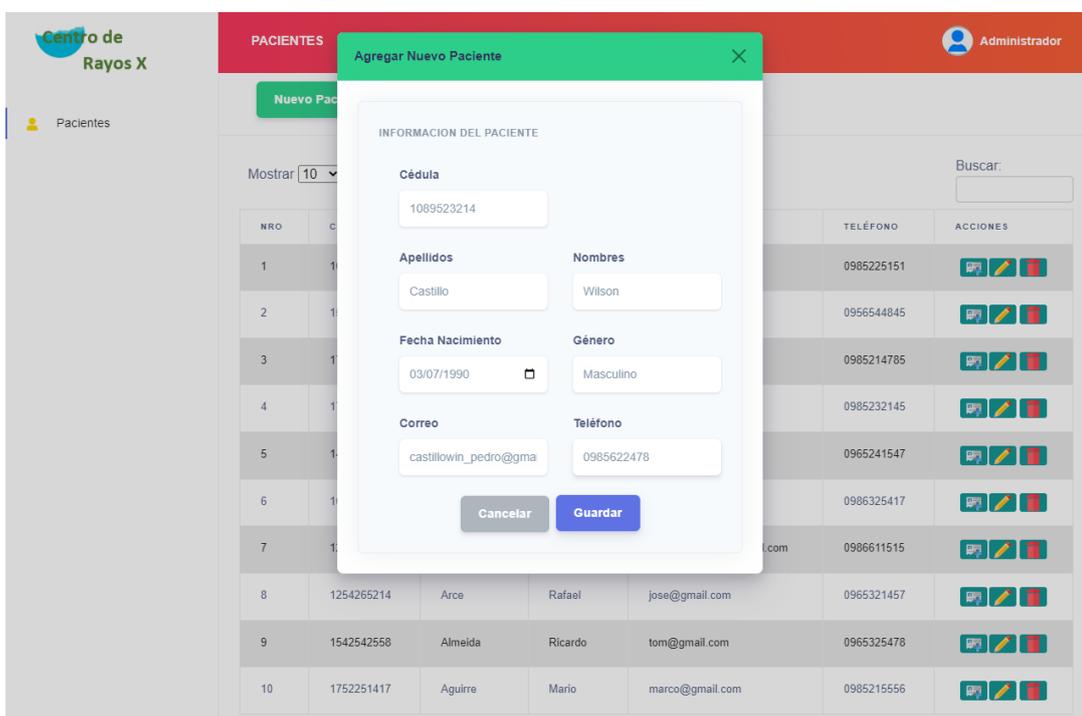


Fig. 55. Agregar nuevo paciente

- c) **Actualizar pacientes:** Los datos anteriormente mencionados pueden ser editados sin ninguna restricción. En la Fig. 56 se visualizado la ventado de edición identificada con otro color.

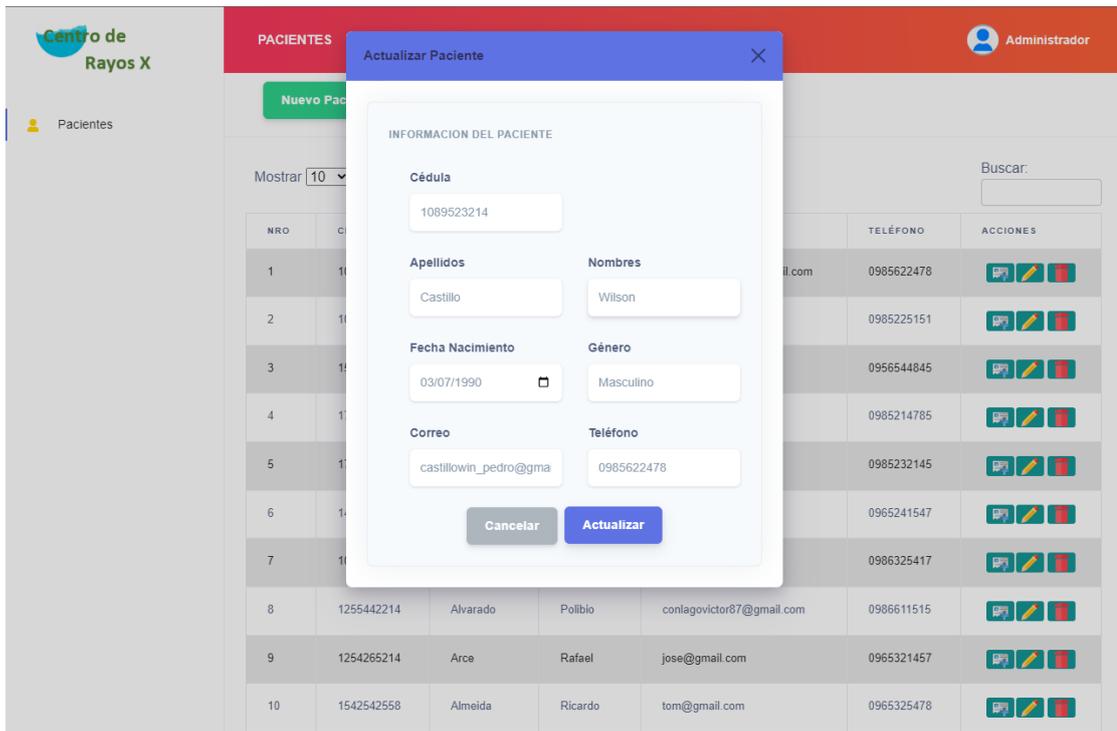


Fig. 56. Actualizar paciente

- d) **Eliminar paciente:** Cuando se elimina un paciente, el registro se bloquea junto con sus referencias en distintas tablas, evitando la pérdida de información permanente. Al momento de eliminar un registro se advierte al usuario si quiere continuar con la acción tal como se muestra en la Fig. 57.

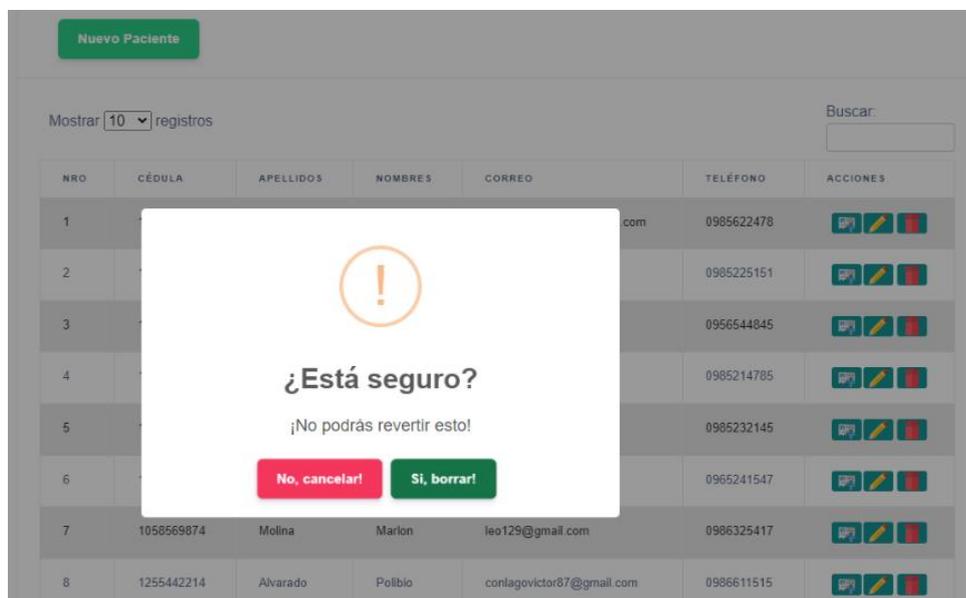


Fig. 57. Eliminar paciente

e) **Listado y consultas del paciente:** Es muy importante contar con un límite de consultas, para evitar la carga excesiva de registros, por ello el usuario puede mostrar una escala de registros. También el aplicativo cuenta con un buscador de forma directa a la base de datos para filtrar por cualquier dato del paciente, tal como se muestra en la Fig. 58.

NRO	CÉDULA	APELLIDOS	NOMBRES	CORREO	TELÉFONO	ACCIONES
1	1089523214	Castillo	Wilson	castillowin_pedro@gmail.com	0985622478	[View] [Edit] [Delete]
2	1008985454	Acosta	David	acosta34@gmail.com	0985225151	[View] [Edit] [Delete]
3	1584465454	Paredes	Marco	marco3@gmail.com	0956544845	[View] [Edit] [Delete]
4	1759652441	Herreria	Fanny	s4fanny@gmail.com	0985214785	[View] [Edit] [Delete]
5	1785425488	Cadena	Damaris	dama94c@gmail.com	0985232145	[View] [Edit] [Delete]
6	1472555278	Bolivar	Dani	dani12hj@gmail.com	0965241547	[View] [Edit] [Delete]
7	1058569874	Molina	Marlon	leo129@gmail.com	0986325417	[View] [Edit] [Delete]
8	1255442214	Alvarado	Polibio	conlagovictor87@gmail.com	0986611515	[View] [Edit] [Delete]
9	1254265214	Arce	Rafael	jose@gmail.com	0965321457	[View] [Edit] [Delete]
10	1542542558	Almeida	Ricardo	tom@gmail.com	0965325478	[View] [Edit] [Delete]

Fig. 58. Límite de consultas y búsqueda

f) **Procesar imagen radiográfica:** En las acciones de cada paciente, se encuentra la opción de ir al apartado de procesamiento de imágenes, en esta parte se tiene que cargar una imagen radiográfica de la columna vertebral, para después ejecutar el análisis haciendo click en el botón azul de tipo flecha, el análisis tomara varios segundos y al final el especialista puede visualizar el resultado, y si es necesario colocar una descripción. En esta parte también cuenta con la opción de editar la imagen y procesarla nuevamente, como también editar la descripción y eliminar el registro.

Con el resultado del ángulo se aprecia a que tipo de clasificación de escoliosis pertenece.

En la Fig. 59 se puede apreciar el resultado final, donde se observa la imagen procesada a partir de la imagen original.

**Centro de Rayos X**

Pacientes

**DASHBOARD** Administrador

**HISTORIAL** Nuevo

FECHA	ACCIÓN
1: 2022-06-27 16:15	👁️ 🗑️
2: 2022-05-04 14:09	👁️ 🗑️
Total : 2	

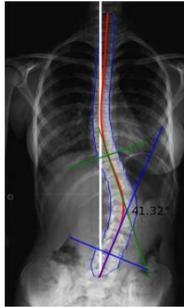
**ANALIZAR IMAGEN**

Actualizar imagen Actualizar

Original



Procesada



Resultado

ÁNGULO	TIPO ESCOLIOSIS
41.32°	Grave

Descripción

Curvatura muy pronunciada en la parte lumbar

**INFORMACIÓN DEL PACIENTE**

Cédula: 1089523214  
 Apellidos: Castillo  
 Nombres: Wilson  
 Fecha Nacimiento: 1990-07-03  
 Género: Masculino  
 Teléfono: 0985622478

Fig. 59. Resultado final con escoliosis de 41.32 grados

Otro ejemplo de escoliosis de tipo grave ubicado en la parte torácica, con un ángulo de 40.12 grados, tal como se muestra en la Fig. 60.

**Centro de Rayos X**

Pacientes

**DASHBOARD** Administrador

**HISTORIAL** Nuevo

FECHA	ACCIÓN
1: 2022-07-22 13:14	👁️ 🗑️
Total : 1	

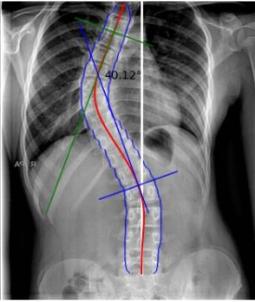
**ANALIZAR IMAGEN**

Actualizar imagen Actualizar

Original



Procesada



Resultado

ÁNGULO	TIPO ESCOLIOSIS
40.12°	Grave

Descripción

Curvatura en la parte torácica

**INFORMACIÓN DEL PACIENTE**

Cédula: 1008985454  
 Apellidos: Acosta  
 Nombres: David  
 Fecha Nacimiento: 1994-12-10  
 Género: Masculino  
 Teléfono: 0985225151

Fig. 60. Resultado final con escoliosis de 40.12 grados

En la siguiente Fig. 61, se observa un diagnóstico de escoliosis con dos curvaturas muy pronunciadas en la parte torácica y lumbar que lamentablemente se corrige con cirugía.

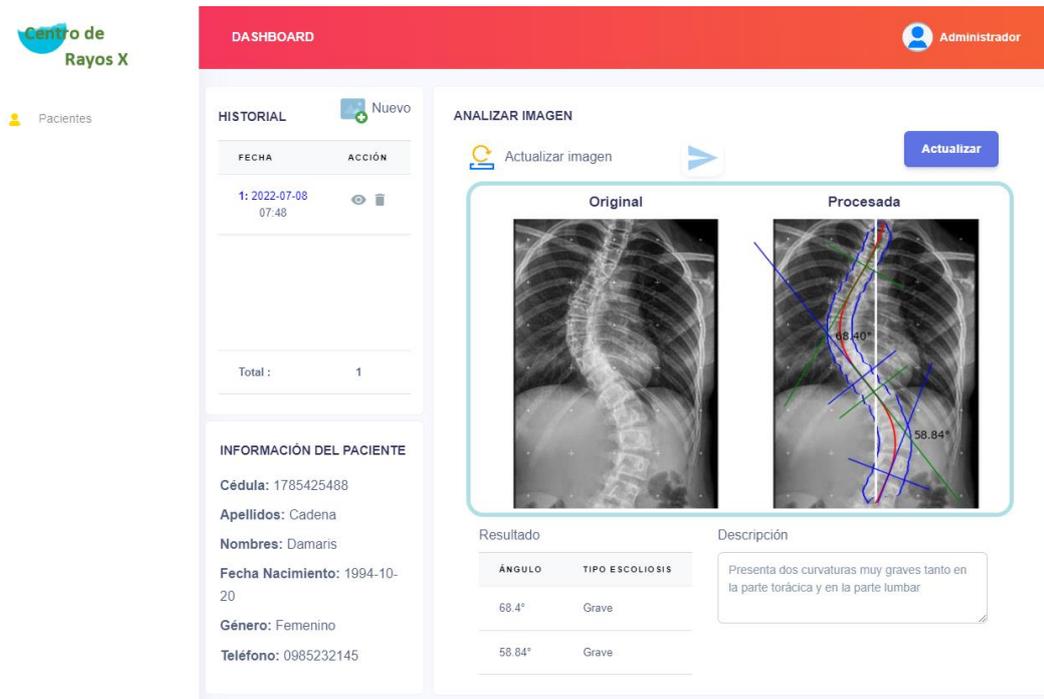


Fig. 61. Resultado final de escoliosis con dos curvaturas de tipo grave

Por último, un resultado de escoliosis de tipo moderada con 30 grados de curvatura.

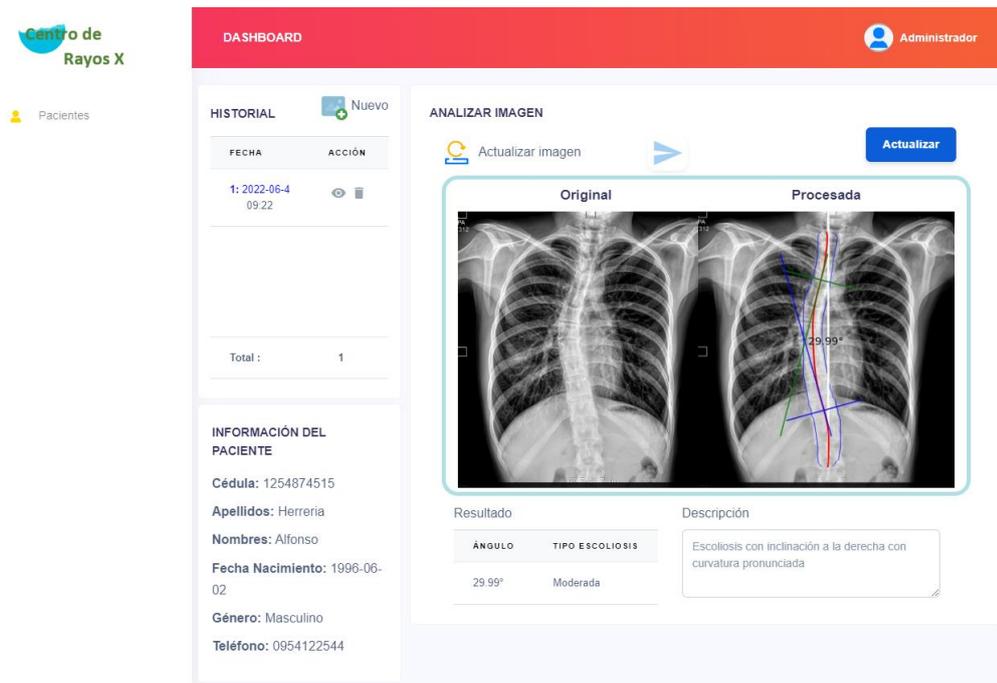


Fig. 62. Resultado final con escoliosis de 30 grados

# CAPÍTULO 3

## Resultados

### 3.1. Análisis de resultados del entrenamiento

En el proceso de entrenamiento se lo realizo de forma local y por Google Colab, esto demostró que es mucho más eficiente entrenar el modelo en Google Colab, pero hay que tomar en cuenta que el servicio que ofrecen es de tan solo 12 horas, a partir de aquí el sistema se reinicia y cualquier información será eliminada. Como alternativa de almacenamiento se hizo uso de Google Drive.

Una vez que se ha completado el entrenamiento con las distintas configuraciones, se obtiene las métricas de la función de pérdida generada por cada época entrenada. La función de perdida evalúa la diferencia entre la salida predicha y la salida deseada, al calcular la diferencia se tiene como resultado un error. Mientras menor sea el resultado de esta función, la red neuronal es más eficiente. En la Fig. 63 se observa la función de pérdida de entrenamiento con un valor de 0.01069.

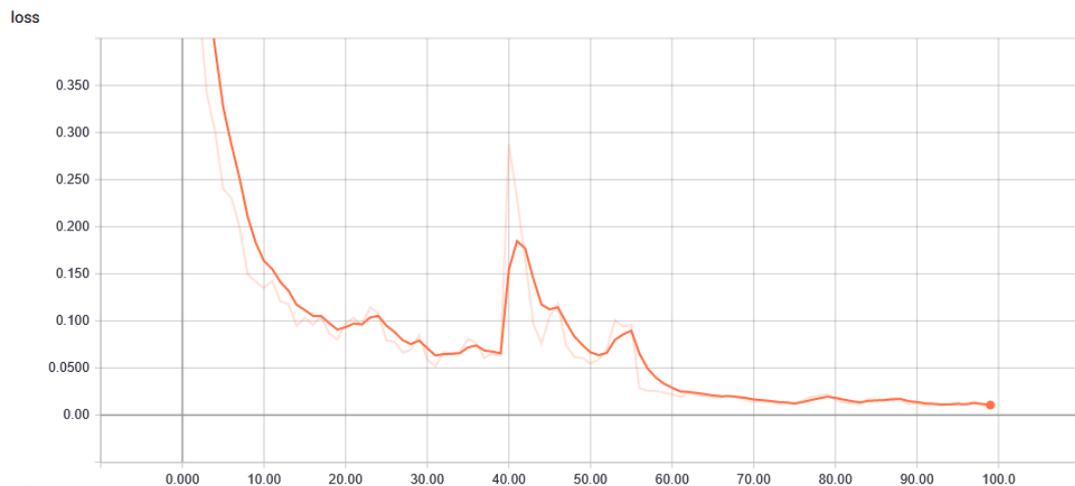


Fig. 63. Pérdida Entrenamiento

En la Fig. 63 se observa un incremento exponencial por la época 40, esto es debido a que se ingresaron nuevos datos para el entrenamiento. Esta parte es conocido como un disparo del error denominado overfitting, el término hace referencia cuando la red neuronal a través de lo aprendido no encaja con un patrón en el proceso del entrenamiento, a causa de este acontecimiento se produce un overfitting.

De igual forma, se presenta en la Fig. 64 las métricas de la función de perdida de evaluación con un valor de 2.093.



Fig. 64. Pérdida Evaluación

### 3.2. Muestreo

Para esta etapa se eligió como muestra 30 radiografías de la columna vertebral de pacientes que padecen la patología de escoliosis, en la cual existe diferentes tipos de curvaturas pronunciadas adecuadas para obtener una variación de resultados.

La medición del grado de desviación de la columna vertebral en imágenes radiográficas fue realizada por 3 personas especialistas, en la Tabla 11 se observa las tres mediciones por cada imagen de la muestra.

Tabla 11. Medición del Ángulo Cobb por especialistas

Nro Radiografía	Observador 1	Observador 2	Observador 3
1	31,00	32,1	33,6
2	9,60	8,1	6,3
3	45,70	44	46,4
4	41,60	45,2	43,7
5	20,60	7,08	24,5
6	6,80	7,4	8,7
7	49,20	45,3	42,5
8	8,70	8,66	7,1
9	50,60	51,3	55,9
10	51,80	52,2	51,3

11	10,30	9,72	8,3
12	30,30	31,7	32,4
13	11,40	8,66	8,5
14	15,30	12,28	17,8
15	41,30	45,7	49,5
16	26,70	29,4	28,7
17	11,00	12,4	14,3
18	19,10	19,2	20,8
19	28,40	25,3	23,6
20	42,10	45,6	52,4
21	54,80	56,3	61,4
22	20,50	16,2	18,3
23	58,20	52,4	40,7
24	7,90	7,6	8,2
25	28,50	32,4	30
26	9,50	9	8,8
27	5,50	6,9	10,1
28	11,60	12	13,2
29	21,30	22,6	23,4
30	29,80	33,2	38,3

---

Una vez obtenido las mediciones calculadas por especialistas, se procedió a realizar un promedio entre los tres ángulos resultantes de cada radiografía, obteniendo como resultado final una sola medición manual.

En esta etapa, se calculó el ángulo de desviación por parte del sistema desarrollado, este cálculo fue realizado en las mismas radiografías de la muestra. A continuación, se visualiza en la Tabla 12 los resultados entre el grado de desviación manual y el grado de desviación del sistema.

Tabla 12. Grado de desviación manual y grados de desviación del sistema

<b>Nro Radiografía</b>	<b>Grado de Desviación Manual</b>	<b>Grado de Desviación del Sistema</b>
1	32,23	29,99
2	8,00	10,56
3	45,37	64,11
4	43,50	42,97
5	17,39	27,97
6	7,63	10,72
7	45,67	41,84
8	8,15	9,65
9	52,60	67,71
10	51,77	59,72
11	9,44	16,21
12	31,47	41,34
13	9,52	12,31
14	15,13	16,13
15	45,50	58,17
16	28,27	25,49
17	12,57	16,03
18	19,70	17,22
19	25,77	23,28
20	46,70	68,40
21	57,50	58,84
22	18,33	22,02
23	50,43	40,12
24	7,90	10,76
25	30,30	27,17
26	9,10	13,61

27	7,50	9,31
28	12,27	10,97
29	22,43	30,98
30	33,77	52,35

Posteriormente, desde la perspectiva de un gráfico al agrupar los resultados obtenidos anteriormente, se puede verificar de mejor manera la gran similitud entre el grado de desviación manual vs el grado de desviación del sistema, los ángulos resultantes obtenidos por las dos perspectivas tienen una diferencia de grados muy pequeña tal como se visualiza en la Fig. 65. Es importante mencionar que las mediciones manuales de cada observador tienen diferentes resultados al momento de localizar la vértebra inicial y final de la curvatura, dando variaciones en la medición del ángulo en una misma radiografía.

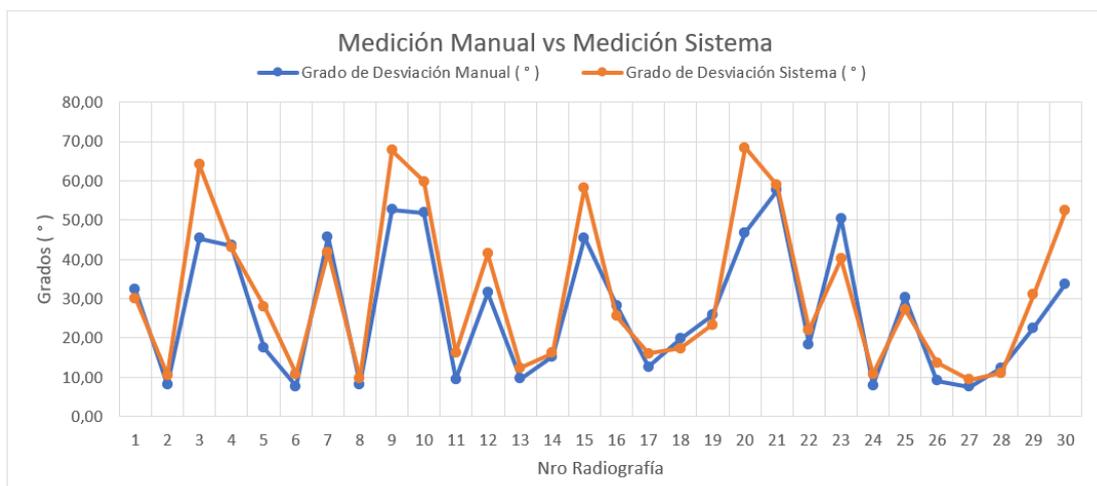


Fig. 65. Medición Manual vs Medición Sistema

Para dar sentido a los datos obtenidos, se tiene que realizar una prueba estadística que demuestra que existe una relación mutua entre las dos mediciones.

En el siguiente trabajo de investigación sobre el tema de “RECONOCIMIENTO DE IMÁGENES PARA DETECCIÓN TEMPRANA DE ALTERACIONES POSTURALES” realizada por la Ing. Claudia Romero, utiliza la prueba t-Student para demostrar la confiabilidad de su sistema desarrollado, lo comprueba mediante el coeficiente de correlación aplicada a dos muestras (Romero Rodríguez, 2019).

### 3.3. Análisis de resultados mediante la prueba t-Student

La prueba t-Student es definida como un tipo de estadística deductiva, permite comparar dos muestras independientes que tengan una distribución normal y homogeneidad en sus varianzas, las muestras deben ser relativamente pequeñas  $n \leq 30$ . La construcción de esta prueba es primero formular la hipótesis nula y la hipótesis alternativa, a partir de ahí calcular la distribución de t, que indica si existe diferencia en la media de las dos muestras independientes y en caso de no existir diferencia, entonces no se rechaza la hipótesis nula y se acepta esta investigación como verdadero (Turcios Sánchez, 2015).

La prueba t-Student posee varios escenarios, entre los más utilizados son:

- **Dos muestras relacionadas o pareadas:** Esta prueba comprara las medias de dos muestras de un solo grupo, que se determina antes del tratamiento y después del tratamiento denominadas medidas pre y post. A partir de las dos variables se calcula las diferencias y se compara si la medida difiere de 0.
- **Dos muestras independientes:** En esta prueba utiliza dos muestras de diferentes grupos o poblaciones, existes dos tipos de pruebas, una con varianzas iguales y la otra con varianzas diferentes. En esta parte se compara las diferencias entre los promedios de las dos poblaciones.

En esta investigación se utilizará la prueba t con dos muestras relacionadas, porque las muestras son obtenidas de la misma población (imágenes de rayos x). La primera muestra es el grado de desviación manual y la segunda muestra el grado de desviación del sistema.

A continuación, se describe las variables y sus fórmulas.

- **Observadores ( $n$ ):** El número de elementos de las muestras.
- **Media ( $\bar{X}$ ):** El promedio de las diferencias entre la primera y segunda muestra.
- **Varianza ( $S^2$ ):** Es una medida de dispersión (variabilidad), se obtiene a partir de las desviaciones al cuadrado de los valores de una muestra con respecto a su media (Francisco López, 2019).

$$S^2 = \frac{\sum_{i=1}^n (X_j - \bar{X})^2}{n-1}$$

- **Desviación estándar ( $S$ ):** Indica la dispersión o variabilidad de un conjunto de datos de una muestra o población, se obtiene al aplicar raíz a la varianza.

$$\sqrt{S^2} = \sqrt{\dots}$$

- **Estadístico ( $t$ ):** Es el número de unidades estándares que indica la diferencia de separación entre las medias de dos muestras, se obtiene a partir de la siguiente fórmula (Turcios Sánchez, 2015).

$$t = \frac{\bar{X}}{\frac{S}{\sqrt{n}}}$$

- **Grados de libertad ( $gl$ ):** Es el número de observaciones que están puramente libres y pueden variar, mediante restricciones matemáticas otorgadas se puede obtener variaciones de características de una población. Los grados de libertad es igual al número de observaciones independientes menos uno (Rodó, 2019).

$$gl = n - 1$$

- **Alfa ( $\alpha$ ):** Denominado como el nivel de significancia, es la probabilidad de rechazar la hipótesis nula cuando la hipótesis planteada es considerada como cierta. El valor de alfa es definido por el investigador, se utiliza con frecuencia alfa = 0.05 o alfa = 0.01 indicando un riesgo de 5% o 1% respectivamente (Chorro Gascó, 2016).
- **p-valor:** Es una probabilidad de que el valor obtenido es producto del azar del muestreo, tomando la hipótesis nula como cierta. El p-valor determina definitivamente si la hipótesis es nula o alternativa, si p-valor es mayor que alfa, no se puede rechazar la hipótesis nula (Data Science Team, 2019).
- **Valor crítico de t:** Es el valor crítico que se compara con el valor absoluto t, si el valor t es menor que el valor crítico, no se puede rechazar la hipótesis nula (Minitab, 2022).
- **Coefficiente de correlación de Pearson:** Es una prueba que mide el grado de covariación entre dos variables de tipo cuantitativos, saber si existe una relación de dependencia lineal entre las dos variables. El coeficiente de correlación de Pearson oscila entre -1 y +1, cuando el valor se acerque a -1 tiene una relación negativa con mucha fuerza, cuando el valor se acerque a +1 tiene una relación positiva con alta asociación y por último cuando una correlación es próxima a 0 no existe relación lineal entre las dos variables (Camacho, 2020).

Para calcular el coeficiente de correlación de Pearson se debe aplicar las siguientes formulas.

- a) El promedio de las variables X y Y.

$$\bar{X} = \frac{\sum X}{n} \quad \bar{Y} = \frac{\sum Y}{n}$$

b) Cálculo de la varianza para la variable X y Y.

$$S_x = \sqrt{\frac{\sum X^2}{n} - \bar{X}^2} \quad S_y = \sqrt{\frac{\sum Y^2}{n} - \bar{Y}^2}$$

c) Cálculo de la covarianza.

$$S_{xy} = \frac{\sum X \cdot Y}{n} - \bar{X} \cdot \bar{Y}$$

d) Por último, cálculo del coeficiente de correlación.

$$r = \frac{S_{xy}}{S_x \cdot S_y}$$

Para dar por concluido la comprobación de resultados al aplicar la prueba t-Student con dos muestras relacionadas, en la Tabla 13 se muestra las diferencias que existe entre el grado de desviación manual vs el grado de desviación del sistema. El coeficiente de correlación de Pearson indica que existe una gran relación entre las dos muestras.

Tabla 13. Prueba t para medias de dos muestras relacionadas

	<b>Grado de Desviación Manual</b>	<b>Grado de Desviación del Sistema</b>
Media	26.8633	31.1983
Varianza	280.0558	391.5015
Desviación estándar	16.73	19.78
Observaciones	30	30
Coeficiente de correlación de Pearson	0.9292	
Diferencia hipotética de las medias	0	
Grados de libertad	29	
Estadístico t	-3.1687	
P(T<=t) una cola	0.00179	

Valor crítico de t (una cola)	1.6991
P(T<=t) dos colas	0.0035
Valor crítico de t (dos colas)	2.0452

También se realizó una prueba t con dos muestras relacionadas, entre observadores para tener en cuenta las diferencias de mediciones del ángulo de la columna vertebral utilizando el método Cobb. En la Tabla 14 se muestra los resultados.

Tabla 14. Prueba t con dos muestras relacionadas entre observadores

	Obs1 vs Obs2	Obs1 vs Obs3	Obs2 vs Obs3
Promedio	0.31	-0.99	-1.29
Correlación	0.9782	0.9526	0.9638
Desviación estándar	16.55	17.06	17.17
t	0.4701	-1.040	-1.5373
p-valor	0.3208	0.1534	0.0675
Valor crítico de t	1.6991	1.6991	1.6991

### 3.4. Confiabilidad del sistema

Como se muestra los resultados en la Tabla 13, la variable coeficiente de correlación de Pearson tiene un valor de 0.9292 demostrando que el sistema es confiable, las dos muestras tienen una relación directa positiva con alta asociación, si el coeficiente fuese próximo a 0 no existiera ninguna relación. En la Fig. 66 se demuestra que la relación es muy fuerte obteniendo una relación lineal.

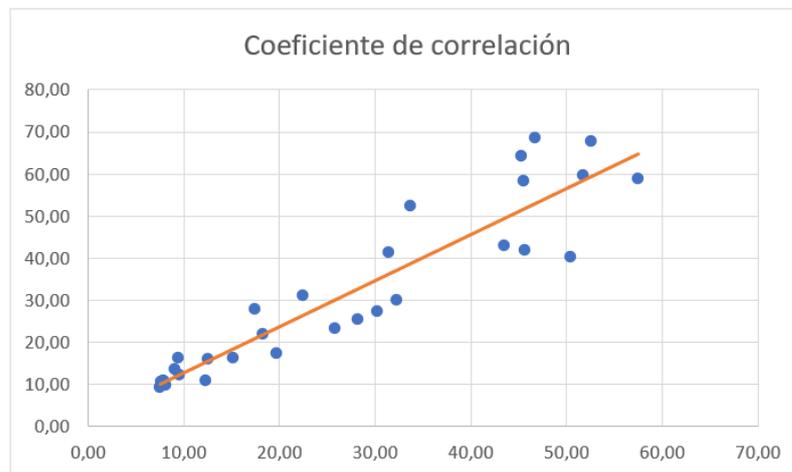


Fig. 66. Coeficiente de correlación

Con respecto a la desviación estándar que indica que tan separados se encuentran los elementos, se calculó la diferencia entre el grado de medición manual de  $16.73^\circ$  y el grado de medición del sistema de  $19.78^\circ$  dando como resultado 3.05 grados de diferencia, la distribución de la medición del sistema no se encuentra muy alejada al compararla con la medición manual.

Finalmente para saber si nuestro resultado se encuentra en la zona de aceptación se tiene que verificar la siguiente condición, donde el valor Estadístico t debe ser menor que el valor crítico. Si se tiene un valor Estadístico t igual a  $-3.1687$  y un valor crítico igual a  $1.6991$ , se considera que nuestra investigación se encuentra en la zona de aceptación al cumplir con la condición.

### **3.5. Discusión**

El presente trabajo de investigación muestra una gran optimización en calcular de forma automática el ángulo de curvatura de la columna vertebral en imágenes radiográficas, tomando alrededor de 15 a 25 segundos de ejecución, el tiempo de ejecución puede ser aún menor si se ocupa una tarjeta gráfica potente. Si se realiza este mismo cálculo de forma manual, el especialista tomara entre 2 a 5 minutos.

Los resultados encontrados en este estudio demuestran que las mediciones de cada observador son distintas, esto varía de acuerdo al criterio del especialista en la selección de vértebras para calcular el ángulo de Cobb. Al realizar la prueba t-Student entre observadores se aprecia que su coeficiente de correlación promedio es de 0.9649, definidas como altamente asociadas, pero su valor no es igual a 1, indicando que existe varianzas a pesar de ser calculadas por especialistas. Si comparamos el coeficiente de correlación promedio de las tres mediciones y el coeficiente de correlación del sistema que tiene un valor de 0.9292, esta comparación de valores indica que aún se puede mejorar el algoritmo para aproximar al cálculo realizado por especialistas.

El trabajo realizado por (Romero Rodríguez, 2019), que de igual manera calcula el ángulo de desviación de la columna vertebral de forma automática, realizó una prueba t-Student con una muestra de  $n=30$ , aplicó un ajuste polinomial de 8 grados y como resultado final obtuvo un coeficiente de correlación de Pearson de 0.8650. Si comparamos con el coeficiente 0.9292 que se obtuvo en este trabajo de investigación, se comprueba que existe una mejora significativa, y al calcular la diferencia entre coeficientes da como resultado 0.064 milésimas, esto indica un gran avance en la automatización y precisión del software realizado.

Otro trabajo realizado con el mismo objetivo de calcular la desviación de la columna vertebral es realizado por (Gómez L & Parra F, 2015), crearon una aplicación para un teléfono inteligente (smartphone), utiliza los sensores de orientación para obtener los datos cuando el

paciente se mueve a petición del especialista y los datos se guardan en la aplicación del teléfono, donde muestra la medición de la desviación de la columna. Para este trabajo es necesario la participación del paciente, no se ocupa imágenes radiográficas a comparación del trabajo de investigación que se desarrolló. Esta app tiene un margen de error de medición del 1.2%, demostrando mayor precisión en calcular el ángulo de desviación de la curvatura de la columna.

Los limitantes del desarrollo de esta investigación es lo complicado en recolectar imágenes radiográficas de la columna vertebral que presenten curvaturas muy pronunciadas, a causa de escasez de casos que presentan esta patología en la cual cuentan con una toma radiográfica.

El software desarrollado en un futuro podría ser subido a un Hosting Web para que cualquier persona pueda analizar imágenes radiografías de pacientes que presentan escoliosis.

El presente trabajo realizado cuenta con la limitación de solo calcular el ángulo de curvatura, sería factible que en futuras investigaciones puedan analizar imágenes realizadas con resonancia magnética, en la cual describan un estudio sobre el estado de la columna vertebral, en este estudio puede incluir el análisis de hernias, deshidratación discal, alineación de los cuerpos vertebrales, etc.

## CONCLUSIONES

- Aplicar regresión polinomial con grados muy elevados la curvatura tiende a justarse drásticamente al modelo perdiendo el sentido de la curva, por ello se realizó varias pruebas para ajustar los datos al modelo de la columna vertebral de forma natural, se consiguió este resultado con 12 grados. Gracias a este modelo se calculó de forma eficiente el ángulo de desviación y al relacionar estos resultados con mediciones por especialistas dio un coeficiente de corrección de 0.92 %, en comparación con otro estudio previo consiguió 0.86 % de correlación aplicando 8 grados en la regresión polinomial, esto significa que el sistema desarrollado obtuvo un 0.06 % de mejora.
- En la actualidad hay sistemas de visión artificial dedicados a la medicina, específicamente a detectar anomalías en imágenes de rayos x del cuerpo humano. Los diferentes estudios han demostrado que la visión artificial detecta con mayor precisión las anomalías radiográficas como el cerebro, pulmones, columna vertebral, entre otros. Esto significa que el ojo humano no es capaz de visualizar colores de píxeles que diagnostican una anomalía.
- Calcular la medición del ángulo de Cobb difiere mucho por cada observador, el resultado depende de la selección de vértebras que conforman la curvatura, dando variaciones en la medición.
- Las librerías de Tensorflow y Keras facilitan la creación de redes neuronales de diferentes arquitecturas, permitiendo desarrollar modelos complejos de aprendizaje automático. Estas dos librerías combinadas son perfectas para adentrarnos en el mundo del Deep Learning gracias a su facilidad de uso y rapidez de ejecución. Cuenta con un selector de clases de optimizadores denominados modelos de aprendizaje profundo, permiten entrenar a la máquina de forma óptima y al terminar el entrenamiento se puede conseguir una tasa de error muy baja.
- Utilizar un modelo pre-entrenado facilita el entrenamiento del nuevo objeto etiquetado, gracias a su amplio aprendizaje en reconocer gran cantidad de características de los diversos objetos. Esto permite reducir el tamaño del dataset, evitando etiquetar miles de imágenes y reduciendo el entrenamiento con cientos de épocas a comparación miles.

## RECOMENDACIONES

- El dataset debe contener imágenes radiográficas con buena resolución y la columna vertebral debe estar completa para detectar el inicio y el final de la desviación de la columna.
- Cada diagnóstico generado por el sistema debe ser respaldado por un especialista antes de tomar cualquier decisión en el tratamiento.
- La arquitectura Mask RCNN para realizar el proceso de entrenamiento, se necesita recursos muy altos de componentes hardware, por ello se recomienda usar ordenadores de altos recursos tanto en procesamiento y memoria. Si se hace uso de la GPU es recomendable que sea mayor igual a 6 GB de memoria RAM, por lo contrario ocupara la CPU para el entrenamiento.
- Para futuras investigaciones se recomienda analizar imágenes de resonancia magnética de la columna vertebral, en estas radiografías se visualiza el estado de la columna y la médula espinal. Es factible desarrollar un sistema en la cual analice y describa un estudio sobre la deshidratación discal, hernia discal, desviación de los cuerpos vertebrales, entre otros.
- Se recomienda, cuando se construye un dataset para visión artificial contar con imágenes de los diversos casos posible del objeto para obtener una mejor predicción y precisión en generar la máscara de interés del objeto.

## GLOSARIO DE TÉRMINOS

**Neurona artificial:** Es un componente primordial en redes neuronales, está inspirado en la neurona del ser humano. Permite retener información importante de una tarea aprendida, si tenemos muchas neuronas conectadas, un conjunto de ellas reconocerá un patrón muy específico.

**Lenguaje de programación:** Lenguaje de programación: es una estructura de código que contiene reglas y al compilarlo hace uso del hardware siguiendo plenamente las instrucciones de cada comando inscripto en un lenguaje de programación como c++, python, java, php, etc.

**Hardware:** Es el componente físico de un circuito impreso, dependiendo del aparato realiza una labor específica.

**Software:** Es el sistema digital de un componente físico, es el controlador del componente. Un programa digital escrito y compilado en un lenguaje de programación hace uso de del hardware para procesar datos y mostrar al usuario mediante gráficos el funcionamiento del sistema.

**Matriz:** Es una estructura de datos que cuenta con columnas y filas, en cada celda se puede almacenar cualquier tipo de información.

**Backend:** Es la parte lógica del desarrollo de una aplicación, es el sistema que controla los datos y las peticiones por parte de clientes, encargado de guardar grandes volúmenes de información y procesarlos.

**Frontend:** Es la parte visual del aplicativo, encargado de contar con una interfaz para el cliente, donde puede realizar tareas correspondientes al aplicativo. Este sistema está conectado a una Api, el cliente mediante la interfaz realiza peticiones al Backend.

**Api:** Es un sistema que escucha peticiones por parte de los clientes y atiende lo solicitado enviando como resultado objetos en formato json, como también imágenes, audio, videos, ficheros, entre otros. Se encuentra ubicado en la parte del Backend.

**Inteligencia artificial (IA):** utiliza redes neuronales conectadas entre sí para aprender una tarea específica previo a un entrenamiento con datos estructurados, La IA utiliza diferentes tipos modelo de aprendizaje para distintas tareas solicitadas.

**Base de datos:** Es una estructura de datos muy organizada que guarda información de manera relacionada entre diferentes registros históricos. Al momento de solicitar un registro su búsqueda es instantánea, se puede elaborar consultas muy complejas filtrando por diferentes registros y obtener un resultado estadístico.

**Dataset:** Es un conjunto de datos que comparten una estructura o un formato específico, podemos encontrar ejemplos de dataset en una hoja de Excel, un folder, archivos almacenados, entre otros.

**CPU:** Es el procesador de la computadora, la unidad central de procesamiento, se encarga de controlar y llevar a cabo cada instrucción solicitada por los diversos programas que contiene un sistema operativo, lo realiza mediante operaciones aritméticas básicas y lógica.

**GPU:** Es la tarjeta gráfica de una computadora, la unidad de procesamiento de gráficos, se encarga de renderizar las imágenes, videos y animaciones mucho más rápido que la CPU.

**Estadística:** Es una ciencia que permite recopilar y estructurar datos para después analizar y estudiar el conjunto de datos con el objetivo de comprender un fenómeno en particular. La estadística pertenece a una rama de las matemáticas.

**Correlación:** Es un tipo de asociación que existe entre dos grupos de variables estadísticas, con el objetivo de saber hasta qué punto las dos variables están relacionadas linealmente teniendo como resultados una tendencia creciente o decreciente.

**Hipótesis:** Es una suposición que puede ser considerada como verdadera, al contar con datos se da inicio a una investigación y ocupar diferentes pruebas estadísticas para confirmar o denegar dicha afirmación.

## REFERENCIAS

- Abdulla, W. (2017). *Mask R-CNN for object detection and instance segmentation on Keras and TensorFlow*. Github. [https://github.com/matterport/Mask\\_RCNN](https://github.com/matterport/Mask_RCNN)
- Alteri, R., & Kalidas, M. (2016, January 20). *Riesgos de la radiación relacionados con los estudios por imágenes*. <https://www.cancer.org/es/tratamiento/como-comprender-su-diagnostico/pruebas/riesgos-de-la-radiacion-asociados-con-los-estudios-por-imagenes.html>
- Angular. (2022). *Angular*. <https://angular.io/>
- Aragón Cervantes, A. A. (2014, May 2). *REGRESION POLINOMIAL – HEFESTO*. <https://bairstown.wordpress.com/category/regresion-polinomial/>
- Bagnato, J. I. (2020, December 24). *Aprende Machine Learning*. Aprende Machine Learning. <https://www.aprendemachinelarning.com/>
- Benages-Pardo, L., Sagüés-Tanco, R., López-Nicolás, G., & Llorente, S. (2019). *Generación de datos sintéticos con objetos de cocina para entrenar redes neuronales de convolución*. 170–177. <https://doi.org/10.17979/spudc.9788497497169.170>
- Blanch Alonso, J. (2021). *Revisión bibliográfica y propuesta de protocolo de actividad física para el tratamiento de la escoliosis lumbar en niños de 12 a 16 años*. <https://repositorio.comillas.edu/xmlui/bitstream/handle/11531/50718/TFG%20CAFYDE%20Blanch%20Alonso%20Javier%20.pdf?sequence=1/>
- Brevity. (2021). *La Inteligencia Artificial*. <https://brevity.pro/wp-content/uploads/2021/10/White-Paper-La-Inteligencia-Artificial.pdf>
- Buhigas, J. (2018, February 14). *Todo lo que necesitas saber sobre TensorFlow, la plataforma para Inteligencia Artificial de Google – Puentes Digitales*. Puentes Digitales. <https://puentesdigitales.com/2018/02/14/todo-lo-que-necesitas-saber-sobre-tensorflow-la-plataforma-para-inteligencia-artificial-de-google/>
- Calvo, D. (2017). *Red Neuronal Convolutacional CNN - Diego Calvo*. <https://www.diegocalvo.es/red-neuronal-convolutacional/>
- Camacho, C. (2020). *COEFICIENTE DE CORRELACIÓN LINEAL DE PEARSON*.
- Casteleiro Villalba, J. M. (2013). *Derivar es fácil - Manual autodidáctico: Vol. Segundo* (Segunda Edición). Business&MarketingSchool.

- Chacón Bustamante, J. M., & Luna Yela, E. P. (2013). *APLICACIÓN DE LA TÉCNICA DE SCHROTH EN PACIENTES CON ESCOLIOSIS QUE ASISTEN AL ÁREA DE REHABILITACIÓN DEL HOSPITAL SAN LUIS DE OTAVALO*.
- Chorro Gascó, J. (2016). *Nivel de significación*. Universidad de Valencia. [https://www.uv.es/webgid/Inferencial/4\\_nivel\\_de\\_significacin.html](https://www.uv.es/webgid/Inferencial/4_nivel_de_significacin.html)
- Creative Commons. (2017, December 2). *Conceptos como clasificación de imágenes, detección de objetos, segmentación semántica, segmentación de instancias, etc.* CSDn.
- Data Science Team. (2019). *Definición del valor P*. Data Science. <https://datascience.eu/es/matematica-y-estadistica/definicion-del-valor-p/>
- DataScientest. (2020, December 16). *Convolutional Neural Network - Deep Learning - DataScientest*. DataScientest. <https://datascientest.com/es/convolutional-neural-network-es>
- Durán Suárez, J. (2017). *Redes Neuronales Convolucionales en R - Reconocimiento de caracteres escritos a mano*.
- Dutta, A. (2018). *VGG Image Annotator*. <https://www.robots.ox.ac.uk/~vgg/software/via/via-1.0.6.html>
- Fernández Ávalos, S. (2011). *La Columna vertebral del niño en crecimiento: desviaciones* (Vol. 30). <https://www.elsevier.es/es-revista-offarm-4-articulo-la-columna-vertebral-del-nino-X0212047X11911437>
- Fernández L, J. (2016). *Recta Tangente y Recta Normal*. <https://www.fisicalab.com/apartado/rectas-tangente-normal>
- Fos Guarinos, B. (2016). *DISEÑO DE TÉCNICAS DE INTELIGENCIA ARTIFICIAL APLICADAS A IMÁGENES MÉDICAS DE RAYOS X PARA LA DETECCIÓN DE ESTRUCTURAS ANATÓMICAS DE LOS PULMONES Y SUS ALTERACIONES*.
- Francisco López, J. (2019). *Varianza - Qué es, definición y significado*. Economipedia. <https://economipedia.com/definiciones/varianza.html>
- Freire, E., & Silva, S. (2019, November 14). *Programa de Visión Artificial/Computacional*. Redes Neuronales. <https://bootcampai.medium.com/redes-neuronales-13349dd1a5bb>
- Gacitúa M, V., González C, M., Sanz, C., Mulli, V., Goddard, P., Rolón D, E., Castello, P., Hiebra, M. D. C., & Pappolla, R. (2016). Adolescent idiopathic scoliosis. *Archivos Argentinos de Pediatría*, 114(6), 585–594. <https://doi.org/10.5546/aap.2016.585>

- García Fontán, J. (2019). *Detección y tracking de objetos mediante la Mask R-CNN*.  
[https://ebuah.uah.es/dspace/bitstream/handle/10017/39327/TFG\\_Garcia\\_Fontan\\_2019.pdf?sequence=1&isAllowed=y](https://ebuah.uah.es/dspace/bitstream/handle/10017/39327/TFG_Garcia_Fontan_2019.pdf?sequence=1&isAllowed=y)
- García Sánchez, E. (2019). *Introducción a las redes neuronales de convolución. Aplicación a la visión por ordenador*. <https://core.ac.uk/download/pdf/290002463.pdf>
- Géron, A. (2017, March). *Training Models - Hands-On Machine Learning with Scikit-Learn and TensorFlow*. <https://www.oreilly.com/library/view/hands-on-machine-learning/9781491962282/ch04.html>
- Gómez L, L., & Parra F, M. (2015). *APLICACIÓN PARA LA DETECCIÓN DE DESVIACIONES DE LA COLUMNA VERTEBRAL APPLICATION FOR THE DETECTION OF VERTEBRAL COLUMN DEVIATION*.
- González Acevedo, J. C., & Rodríguez Pérez, J. C. (2016). Nuevo concepto de unidad funcional lumbo-sacroccígea- pélvica. Bases teóricas y repercusión en el análisis clínico y terapéutico de los pacientes con dolor lumbar. In *Revista de la Sociedad Espanola del Dolor* (Vol. 23, Issue 5, pp. 260–268). Ediciones Doyma, S.L. <https://doi.org/10.20986/resed.2016.3422/2016>
- Gonzalo, Á. (2018). *Tipos de aprendizaje automático*. MACHINE LEARNING PARA TODOS. <https://machinelearningparatodos.com/tipos-de-aprendizaje-automatico/>
- Gutiérrez, D. (2021). INTRODUCCIÓN A LAS REDES NEURALES PARA VISIÓN ARTIFICIAL CON PYTHON Y RASPBERRY PI. *ResearchGate*, 39. <https://doi.org/10.13140/RG.2.2.20121.34409>
- He, K., Zhang, X., Ren, S., & Sun, J. (2015). *Deep Residual Learning for Image Recognition*. <http://image-net.org/challenges/LSVRC/2015/>
- He Kaiming, Gkioxari Georgia, Dollár Piotr, & Girshick Ross. (2018, January 24). *Mask R-CNN*. 12. <https://arxiv.org/abs/1703.06870>
- Hernán Mondragón, O., Perafán, J., & Mayor, W. (2015). *Visión vial Inteligencia. Análisis de patrones de accidentalidad*. 283.
- Herrera, P. J., Guijarro, M., & Guerrero, J. M. (2018). *Operaciones de Transformación de Imágenes*. <https://www.researchgate.net/publication/323200190>
- IBM. (2022). *¿Qué es la Visión Artificial?* IBM. <https://www.ibm.com/es-es/topics/computer-vision>

- IONOS. (2020, October 8). *¿Qué es Keras? Introducción a la biblioteca de redes neuronales*. Digital Guide. <https://www.ionos.es/digitalguide/online-marketing/marketing-para-motores-de-busqueda/que-es-keras/>
- Juan, I., Lozano, L., Dávila, C., Mora, J., & Tramontini, C. (2018). Anatomía de la columna vertebral en radiografía convencional. *Revista Médica Sanitas*, 21(1), 39–46. <https://doi.org/10.26852/01234250.11>
- Kurbiel, T. (2021, April 22). *Derivative of the Softmax Function and the Categorical Cross-Entropy Loss*. Towards Data Science. <https://towardsdatascience.com/derivative-of-the-softmax-function-and-the-categorical-cross-entropy-loss-ffceefc081d1>
- Landa, M. (2018). *MASTER'S THESIS MASK R-CNN IN GRASS GIS*.
- Lin, T.-Y., Dollár, P., Girshick, R., He, kaiming, Hariharan, B., & Belongie, S. (2017, April 19). Feature Pyramid Networks for Object Detection. *Facebook AI Research (FAIR)*, 10. <https://arxiv.org/abs/1612.03144v2>
- Lisintuña Molina, J. X. (2015). *PREVALENCIA DE ESCOLIOSIS DE COLUMNA LUMBAR, EN PACIENTES DE 45 A 70 AÑOS CON RAYOS X DIGITAL, EN LA CLÍNICA CRUZ BLANCA DE LA CIUDAD DE QUITO*.
- Martínez Ojeda, G. E. (2022). DESARROLLO DE UN CLASIFICADOR BASADO EN REDES NEURONALES PARA LA DETECCIÓN DE ESCOLIOSIS EN IMAGEN RX DE COLUMNA. *UAN*, 90. [http://repositorio.uan.edu.co/bitstream/123456789/6085/3/2022\\_Trabajo.G.Martinez%2CGonzalo.pdf](http://repositorio.uan.edu.co/bitstream/123456789/6085/3/2022_Trabajo.G.Martinez%2CGonzalo.pdf).
- Minitab. (2022). Uso del valor t para determinar si se puede rechazar la hipótesis nula. *Soporte de Minitab*.
- Miranda, F. (2017). *Generalidades de la columna Vertebral*. <https://www.mirandafisioterapia.com/post/2017/08/16/generalidades-columna-vertebral>
- Moreno Díaz, A. L. (2020). *Análisis comparativo de arquitecturas de redes neuronales para la clasificación de imágenes [Inteligencia Artificial]*. Máster Universitario en Inteligencia Artificial.
- Ocaña, J. (2013). *Gestión de proyectos con mapas mentales: Vol. Vol. 1* (Club Universitario).
- Oliveira, C., García Navarro, R., Ruiz Caballero, J. A., & Brito Ojeda, E. (2011). *Biomecánica de la columna vertebral*.

- Pérez, P., & Valente, M. (2018). *Fundamentos básicos del procesamiento de imágenes*. Curso de Imágenes Médicas. <https://www.famaf.unc.edu.ar/~pperez1/manuales/cim/cap2.html>
- Pulido Romero, E., Escobar Domínguez, Ó., & Núñez Pérez, J. Á. (2019). *Base de datos* (Vol. 0). Grupo Editorial Patria. <https://elibro.net/es/lc/utnorte/titulos/121283>
- Raneros, R. R. (2021). *Estudio de la arquitectura YOLO para la detección de objetos mediante deep learning*. <https://uvadoc.uva.es/bitstream/handle/10324/45359/TFM-G1316.pdf?sequence=1>
- Rathnayaka, Igd., & Kumara, B. (2020). *A Review of Software Development Methodologies in Software Engineering* (Vol. 6). [http://ijariie.com/AdminUploadPdf/A\\_Review\\_of\\_Software\\_Development\\_Methodologies\\_in\\_Software\\_Engineering\\_ijariie12553.pdf](http://ijariie.com/AdminUploadPdf/A_Review_of_Software_Development_Methodologies_in_Software_Engineering_ijariie12553.pdf)
- Recuero de los Santos, P. (2021, December 2). *Machine learning: conoce qué es y las diferencias entre sus tipos*. Telefónica Tech. <https://empresas.blogthinkbig.com/que-algoritmo-elegir-en-ml-aprendizaje/>
- Rodó, P. (2019). *Grados de libertad*. Economipedia. <https://economipedia.com/definiciones/grados-de-libertad.html>
- Romero Rodríguez, C. (2019). *RECONOCIMIENTO DE IMÁGENES PARA DETECCIÓN TEMPRANA DE ALTERACIONES POSTURALES*.
- Sánchez Alor, J. (2020). *Evaluación de algoritmos de detección de objetos basados en deep learning para detección de incidencias en carreteras*. 117. <https://uvadoc.uva.es/bitstream/handle/10324/43277/TFG-G4450.pdf?sequence=1&isAllowed=y>
- Sánchez García, A. O., & Méndez Durán, R. (2015). *MEDIDAS RADIOGRÁFICAS HABITUALES EN COLUMNA VERTEBRAL*. [https://unitia.secot.es/web/manual\\_residente/CAPITULO%2063.pdf](https://unitia.secot.es/web/manual_residente/CAPITULO%2063.pdf)
- Sánchez Márquez, P. (2016). *ANÁLISIS RADIOLÓGICO DE LOS ESCOLIOGRAMAS*. [https://www.serme.es/wp-content/uploads/2017/01/analisis\\_radiologico\\_escoliograma1.pdf](https://www.serme.es/wp-content/uploads/2017/01/analisis_radiologico_escoliograma1.pdf)
- SBV ORTOPEDIA. (2019). *¿Todo lo que debes saber acerca de la escoliosis?* - SBV. <http://sbvortopedia.com/que-es-la-escoliosis/>
- Serra Requena, B. (2021, March). *Ángulo entre dos rectas*. Universo Formulas. <https://www.universoformulas.com/matematicas/geometria/angulo-dos-rectas/>

- Srivastava, S., Vishvas Divekar, A., Anilkumar, C., Naik, I., Kulkarni, V., & v, P. (2020). *Comparative Analysis of Deep Learning Image Detection Algorithms*. <https://doi.org/10.21203/rs.3.rs-132774/v1>
- Timarán Pereira, S. R., Hernández Arteaga, I., Caicedo Zambrano, S. J., Hidalgo Troya, A., & Alvarado Pérez, J. C. (2016). El proceso de descubrimiento de conocimiento en bases de datos. In *Descubrimiento de patrones de desempeño académico con árboles de decisión en las competencias genéricas de la formación profesional*. Universidad Cooperativa de Colombia. <https://doi.org/10.16925/9789587600490>
- Torres Alonso, A. (2020). *DETECCIÓN DE FRUTAS EN ÁRBOLES*.
- Turcios Sánchez, A. R. (2015). *t-Student. Usos y abusos*. <http://www.medigraphic.com/revmexcardiol>
- Utrera Burgal, J. (2018, December 5). *Deep Learning básico con Keras (Parte 4): ResNet*. <https://www.enmilocalfunciona.io/deep-learning-basico-con-keras-parte-4-resnet/>

## ANEXOS

**Anexo 1.** Anotador de imágenes VGG: <https://bit.ly/3fuuQ4e>

**Anexo 2.** Mask R-CNN master: <https://bit.ly/3Co75nD>