

Desarrollo de un Algoritmo de Segmentación para la Detección de Espacios Interdentales en Radiografías Panorámicas Dentales

Development of a Segmentation Algorithm for Detecting Interdental Spaces in Panoramic Dental Radiographs

Moreno Salgado Ana Goretti¹, Alcaraz Nuñez Mariana Paola¹, Navarro Meza Víctor Hugo², López Márquez Miguel Ángel¹, Francisco Javier Cuevas de la Rosa³

¹División de Ciencias e Ingenierías, Campus León, Universidad de Guanajuato.

²Unidad Profesional Interdisciplinaria de Ingeniería, Campus Guanajuato, Instituto Politécnico Nacional.

³Centro de Investigaciones en Óptica, A.C. León, Gto.

ag.morenosalgado@ugto.mx, mp.alcaraznunez@ugto.mx, vnavarrom2101@alumno.ipn.mx, ma.lopezmarquez@ugto.mx, fjcuevas@cio.mx

*Autor de correspondencia

Resumen

Las radiografías panorámicas dentales son una herramienta clave en el diagnóstico odontológico, sin embargo, su interpretación puede verse afectada por factores como la superposición anatómica y la variabilidad entre pacientes. Por ello, el desarrollo de algoritmos de segmentación automatizada representa una alternativa prometedora para apoyar el análisis clínico. El presente trabajo describe el desarrollo de un algoritmo en lenguaje Python de segmentación orientado a la detección automática de espacios interdentes en radiografías panorámicas dentales. Se realizaron cuatro versiones del algoritmo, utilizando un enfoque de bajo nivel y con uso mínimo de bibliotecas externas, lo cual permitió controlar cada etapa del procesamiento. Los códigos resultantes fueron probados sobre una radiografía representativa, mostrando resultados consistentes y precisos en la detección de diastemas. Este trabajo representa un avance en herramientas más accesibles de apoyo clínico y una base para futuros sistemas automatizados en diagnóstico odontológico.

Palabras clave: segmentación de imágenes médicas; radiografías panorámicas dentales; procesamiento digital de imágenes; espacios interdentes.

Introducción

La odontología es una rama de la medicina que estudia la anatomía, el desarrollo y las enfermedades de los dientes, lo que la convierte en un importante campo de investigación. Existen múltiples problemas dentales cuya detección temprana es crucial para prevenir complicaciones y conservar la calidad de vida del paciente. Una de las técnicas de imagen médica más utilizadas en el diagnóstico es la radiografía dental [1].

La radiografía consiste en el registro fotográfico de una imagen producida por el paso de una fuente de rayos X a través de un objeto. En odontología, esta herramienta se usa para evaluar el estado de los dientes, las encías, los maxilares y la estructura ósea de la boca. Con esta información es posible detectar las causas en etapas tempranas y proponer tratamientos óptimos para los pacientes. Otra aplicación de las radiografías dentales es en la odontología forense, la cual pretende identificar a los individuos basándose en sus características dentales [2]. Sin embargo, la calidad de la imagen radiográfica puede verse afectada por diversos factores como el dispositivo utilizado, la técnica de adquisición y su posterior procesamiento. Una de las limitaciones de este tipo de imágenes es que es una representación bidimensional de un objeto tridimensional, por lo que existe una superposición de las estructuras anatómicas. A este problema se le suman otras dificultades como las variaciones anatómicas entre pacientes, artefactos utilizados para restauraciones y prótesis, espacio existente por un diente ausente, entre otros [2,3]. Todo esto hace que el desarrollo de herramientas informáticas automatizadas de ayuda al diagnóstico odontológico sea un desafío en la actualidad.

Ante esta situación, el procesamiento digital de las radiografías mediante técnicas de segmentación automatizada puede ser una solución prometedora. La segmentación es un proceso en el que se divide una imagen en regiones u objetos para simplificar y facilitar su análisis. Cuando se realiza de manera robusta permite identificar de forma clara las distintas estructuras presentes en la imagen, lo que contribuye significativamente a resolver los problemas derivados de la baja calidad, la superposición de elementos o la variabilidad anatómica entre pacientes [4]. Diferentes métodos de segmentación han sido utilizados para abordar distintos problemas dentro del área odontológica con múltiples beneficios para el profesional de la salud y el paciente. Existen casos en los que el diagnóstico asistido por computadora se ha empleado de manera eficiente; por ejemplo, un estudio se centró en la detección automática de fracturas mandibulares mediante análisis de textura, con el objetivo de identificar posibles casos de osteoporosis y otras anomalías en la mandíbula [5]. En otro trabajo, se propuso un método de segmentación basado en franjas verticales, diseñado para aislar de forma automática la mandíbula inferior y el hueso mandibular en radiografías panorámicas digitales [6].

Por lo que, la presente investigación tiene como objetivo desarrollar un algoritmo de segmentación para la detección de espacios interdentes en radiografías panorámicas dentales en colaboración con el Centro de Investigaciones en Óptica (CIO), como un primer avance hacia algoritmos más específicos.

Definiciones Matemáticas

Operaciones morfológicas

Reducción de resolución

La interpolación bilineal es un método de reducción de imágenes que estima nuevos valores mediante el promedio ponderado de los niveles de gris de cuatro píxeles vecinos, aplicando una función bilineal en las coordenadas. Esta estimación permite generar una imagen reducida $g(n1, n2)$ con menor resolución según un factor de reducción predefinido [7,8] mediante el cálculo de:

$$g(n1, n2) = \frac{A_{00} + A_{01} + A_{10} + A_{11}}{4} \quad (1)$$

Si se aplica un factor r , la nueva imagen tiene tamaño $M/r \times N/r$. Reducir la resolución de la imagen permite trabajar con menos datos y acelera el procesamiento, priorizando eficiencia sobre detalle. [4]

Escala de grises

Si una imagen RGB se representa como una función:

$$I(x, y) = [R(x, y), G(x, y), B(x, y)] \quad (2)$$

RGB

La conversión a escala de grises se define como la combinación lineal ponderada de los canales rojo, verde y azul:

$$I_{gr}(x, y) = 0.2989 * R(x, y) + 0.5870 * G(x, y) + 0.1140 * B(x, y) \quad (3) [9][10]$$

Binarizado

Esta operación crea una imagen binaria $B(i, j)$ a partir de una imagen de grises $I(i, j)$. Esto se realiza estableciendo un umbral fijo U y partir de una función que asigna a cada píxel de la imagen sólo dos valores posibles. [11]

$$B(i, j) = \begin{cases} 1, & \text{si } I(i, j) \geq U \\ 0, & \text{si } I(i, j) < U \end{cases} \quad (3)$$

Erosión

La erosión reduce el área al eliminar o desgastar los píxeles en los bordes del objeto. Es decir, el conjunto de todos los puntos z , tales que el elemento estructurante B , trasladado por z , queda completamente contenido dentro de A . [12] Puede representarse como:

$$A \ominus B = \{z | B_z \subseteq A\}, \quad (4)$$

La dilatación expande los objetos en una imagen binaria, agregando píxeles a los bordes, se define formalmente como [4]:

$$A \oplus B = \{z | (B_z) \cap A \neq \emptyset\}, \quad (5)$$

Es decir, la dilatación de un conjunto X es el conjunto de puntos donde el elemento estructurante B , al reflejarse y desplazarse, toca a X [13]; su efecto depende de la forma y tamaño de B , elegidos según los detalles que se deseen modificar. [14]

Centroides

El centroide (x, y) de una región se define como:

$$x = \frac{1}{A} \sum_{x,y} x \cdot B(x, y), \quad y = \frac{1}{A} \sum_{x,y} y \cdot B(x, y), \quad (6)$$

donde A es el área (número de píxeles que forman la región) y B se define como: [13] [15]

$$B(x, y) = \begin{cases} 1 & \text{si el píxel } (x, y) \text{ pertenece a la región} \\ 0 & \text{si no pertenece} \end{cases} \quad (7)$$

Región intramaxilar

La región intramaxilar suele hacer referencia a el área cerrada y delimitada por la disposición de los dientes, donde no se observan estructuras óseas dentales, y que puede analizarse para buscar anomalías, simetrías o segmentaciones dentro de radiografías panorámicas o intraorales.[16] Se caracteriza por que es más oscura que el resto de la imagen y se ubica entre el 40% y el 60% de la altura de las panorámicas dentales.[17]

Simetrización

La simetrización de una región R (conjunto de píxeles en una imagen binaria) consiste en modificarla para que sea simétrica respecto a un eje dado, generando una nueva región R_s . La simetría por reflexión respecto a una línea vertical

$x = c$ se define como: $(x, y) \in R \Rightarrow (2c - x, y) \in R_s$, (8) para después pasar a la simetrización total de una región R que se define: $R_s = R \cup \text{Reflexión}(R)$, (9). Es decir, se construye la imagen reflejada de R respecto al eje vertical $x = c$ y se combina con la región original. [18]

Línea Central

La línea central de la región intramaxilar, se obtiene calculando el píxel central de cada columna de píxeles que se encuentra en dicha región. Este conjunto de datos se usan como apoyo para el ajuste de la spline. [17]

Spline

Se refiere a una regresión polinómica utilizada para suavizar los datos de la línea central obtenida de la región intramaxilar, utilizando cierto número de puntos equidistantes y dando como resultado un polinomio de la forma [19]:

$$P_m(x) = \frac{0}{m}x^0 + \frac{1}{m}x^1 + \frac{2}{m}x^2 + \dots + \frac{m}{m}x^m \quad (10)$$

División de la spline

Se refiere a la acción de segmentar el polinomio ajustado (spline) en intervalos definidos por puntos equidistantes, lo que permite analizar o aplicar el suavizado de manera localizada y controlada sobre la línea central intramaxilar.

Espacios interdentes

Los espacios interdentes son áreas situadas entre dientes adyacentes que, en imágenes radiográficas, se manifiestan como regiones de baja densidad o intensidad. Su presencia, ausencia o alteración puede indicar condiciones normales, patologías, o requerimientos ortodónticos. Representan regiones clave para la segmentación y el análisis automatizado en radiografías dentales. Al ser detectados, permite mejorar la delimitación de piezas dentales, identificar diastemas, evaluar la alineación dental y apoyar el diagnóstico clínico con herramientas computacionales. [20]

Complemento

La imagen complemento es aquella que se obtiene al invertir los valores de intensidad de cada píxel de una imagen.

Matemáticamente, si una imagen original se representa como $I(x, y)$, su complemento se define como: $I_{complemento}(x, y) = 255 - I(x, y)$, (11). Se utiliza frecuentemente para facilitar la segmentación, resaltar objetos oscuros sobre fondo claro o viceversa, y en operaciones morfológicas como la identificación de componentes conectados o la detección de bordes. [4]

Operadores de Sobel

Los operadores de Sobel tienen la tarea de suavizar la imagen de tal manera que se elimina un poco de ruido de la imagen si es que lo tiene, por lo consiguiente se puede desaparecer falsos bordes. [21] Se representan como dos matrices de 3x3, las cuáles se definen como:

$$S_x = \begin{pmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{pmatrix} \quad S_y = \begin{pmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{pmatrix} \quad (12)$$

Permiten aproximar las derivadas parciales de una imagen arbitraria y, a partir de dichos valores, se puede calcular la magnitud del gradiente, y así extraer el contorno de los objetos en una imagen. [18] usando la ecuación:

$$\|\nabla I(x, y)\| = \sqrt{I_x^2 + I_y^2} \quad (13)$$

Metodología

El algoritmo completo consta de ocho pasos, mismos que se anudan uno a uno para obtener como resultado final la identificación de los espacios interdentes en las radiografías panorámicas dentales:

Se desarrollaron cuatro versiones diferentes del algoritmo completo, cada uno con distinta lógica de programación y diferentes herramientas utilizadas. Pese a que los cuatro siguen los mismos pasos para llegar a un objetivo final común, puede que tanto sus tiempos como sus salidas no sean las mismas para cada paso, es por eso que en el desglose se muestra la ejecución y lógica de cada uno, así como la teoría detrás de cada paso y el tiempo de procesamiento del código completo. Cada código se probó en una muestra de 5 a 10 imágenes obtenidas de una serie de pacientes de la Clínica de Odontología de La Salle Bajío en la ciudad de León, Guanajuato, México, gracias a la colaboración de la Dra. Miriam Rocha de la Facultad de Odontología de la Universidad de la Salle Bajío. La imagen con la que se obtendrán la comparativa en los códigos se muestra en la figura 2. Asimismo, el código fué desarrollado en el lenguaje de programación python, con un uso de mínimo a nulo de librerías especializadas.

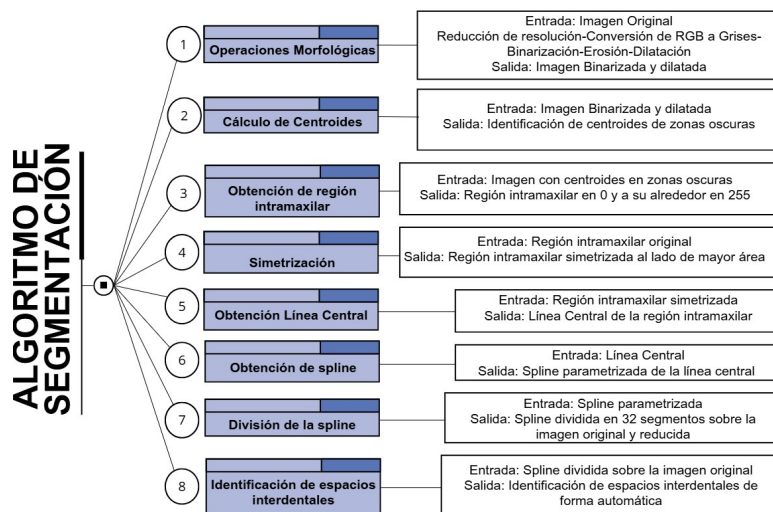


Figura 1. Algoritmo general.



Figura 2. Radiografía base.

Resultados y Discusiones

A continuación se desarrollará y explicará cada una de las lógicas de programación que se siguieron para completar los objetivos de cada paso del algoritmo general:

Operaciones morfológicas

Reducción de resolución de la imagen

De acuerdo con la definición *reducción de resolución* se desarrollan cuatro diferentes lógicas de programación que se explican a continuación:

Algoritmo a (Fig 3.a). La imagen original tiene una resolución de 1202 x 645 píxeles pasa a 300 x 161 píxeles, lo que significa que tuvo un factor de reducción de 4. Esto para evitar la reescalación de la imagen y optimizar el procesamiento del programa, manteniendo una resolución aceptable, sin perder información necesaria para las siguientes etapas.

Algoritmo b (Fig 3.b). En este algoritmo se determinó un factor de reducción de 16, con el fin de comparar la velocidad de los primeros pasos si se tiene una imagen de baja resolución al inicio, para luego ser reescalada más adelante.

Algoritmo c (Fig 3.c). Para este código se utilizó un factor de reducción de 4, con la finalidad de facilitar el procesamiento de la imagen y reducir el tiempo de las operaciones posteriores, al final de las cuales, la imagen se restablecerá a su resolución original. La resolución original era de 1935 x 1024 píxeles y se redujo a 484 x 256 píxeles.

Algoritmo d (Fig 3.d). La redimensión de imágenes se logra al dividir la imagen en bloques (definidos por un factor de reducción, por ejemplo, 2x2 píxeles), promediar los valores de píxel en cada bloque, y usar esos promedios para crear una nueva imagen de menor resolución. Finalmente, se comparan la imagen original y la reducida.

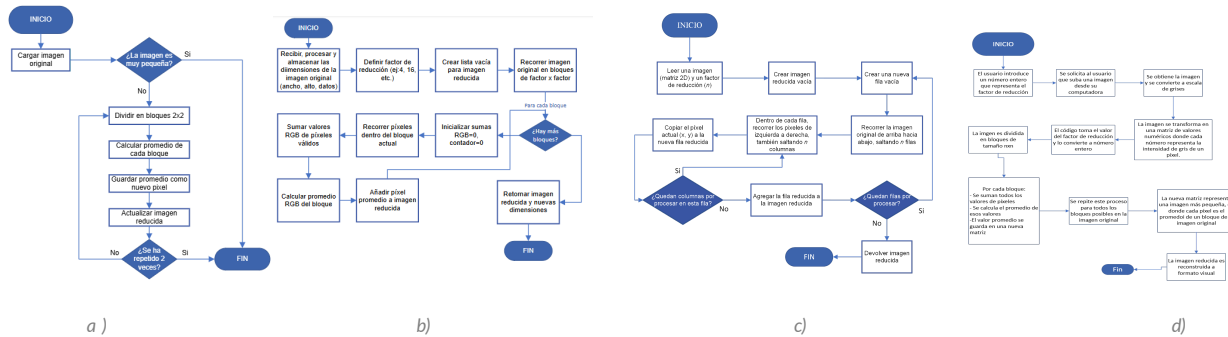


Figura 3. Algoritmos para reducción.

Escala de grises y su histograma

De acuerdo con la definición *escala de grises* se desarrollan cuatro diferentes lógicas de programación que se explican a continuación:

Algoritmo a (Fig 4.a). Se localiza la intensidad umbral del 40% de la suma acumulada, en este caso, el umbral es de 86.

Algoritmo b (Fig 4.b). Se calculó el histograma simultáneamente durante este proceso para optimizar recursos. Esto permite simplificar los procesos posteriores y analizar la distribución de tonos de gris antes de la binarización, clave para determinar umbrales adaptativos más adelante. El umbral resultante fue de 86.

Algoritmo c (Fig 4.c). A partir de este algoritmo, se cambió la información de los píxeles de la imagen RGB a la escala de grises con valores de 0 a 255. Además, se generó un histograma con estos datos, para determinar el umbral al que se realizará la operación de binarización (debajo del cual están el 45% de los píxeles totales de la imagen).

Algoritmo d (Fig 4.d). Se construye una función de transformación acumulativa a partir de estas probabilidades normalizadas. Esta función redistribuye los niveles de gris, actuando como la base de la ecualización. Finalmente, cada píxel de la imagen original es reemplazado por su nuevo valor transformado usando esta función. El resultado es una nueva imagen con un contraste mejorado.

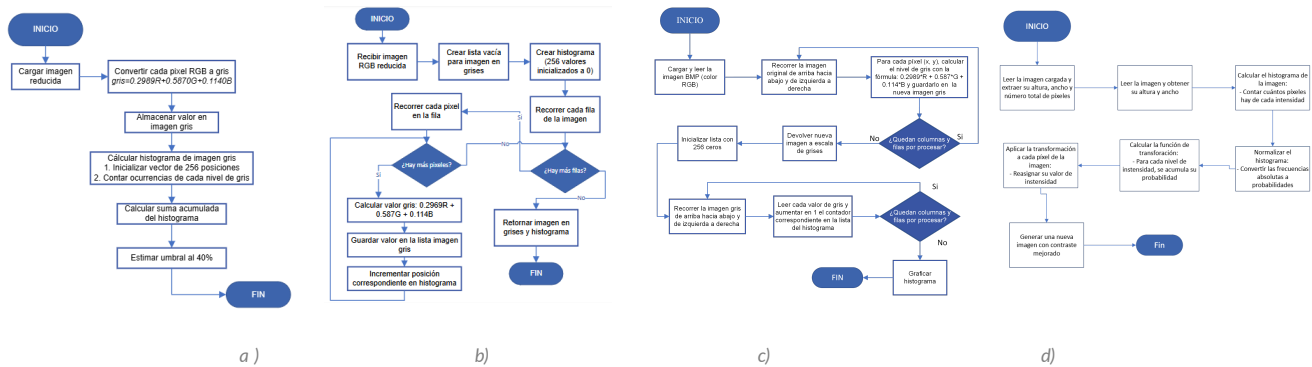


Figura 4. Algoritmos para convertir una imagen RGB a escala de grises.

Binarizado e histograma

De acuerdo con la definición *binarizado* se desarrollan cuatro diferentes lógicas de programación que se explican a continuación:

Algoritmo a (Fig 5.a). Cuálquier píxel con intensidad mayor a 86 se coloca en blanco y los píxeles menores o iguales a 86 se colocan en negro, generando la binarización de la imagen. Luego, a modo de comprobación, se calcula un histograma binario que muestra la cantidad de píxeles blancos y negros, obteniendo el 40.96% de píxeles en negro y el 59.04% de píxeles en blanco, lo que significa un error porcentual del 2.4%

Algoritmo b (Fig 5.b). El binarizado no usa un umbral fijo, sino dinámico según el porcentaje de área deseada. Esto garantiza adaptabilidad a variaciones de iluminación. El histograma binario de 2 niveles (0/255) sirvió como control de calidad, comparando la cantidad de píxeles 0 (3029) y 255 (4651), verificando que la segmentación mantuviera la proporción esperada entre objeto y fondo.

Algoritmo c (Fig 5.c). A partir de este código, se cambiaron los valores de los píxeles a blanco y negro, de modo que el 45% de los píxeles se encontraran en 0 (negro) y poder identificar mejor las regiones de la imagen.

Algoritmo d (Fig 5.d). Se convierte la imagen a escala de grises, se calcula el histograma de intensidades y se determina un umbral automático (30% de píxeles más oscuros). Este umbral se usa para binarizar la imagen a blanco y negro, y finalmente, se visualizan la imagen binarizada y su histograma B/N.

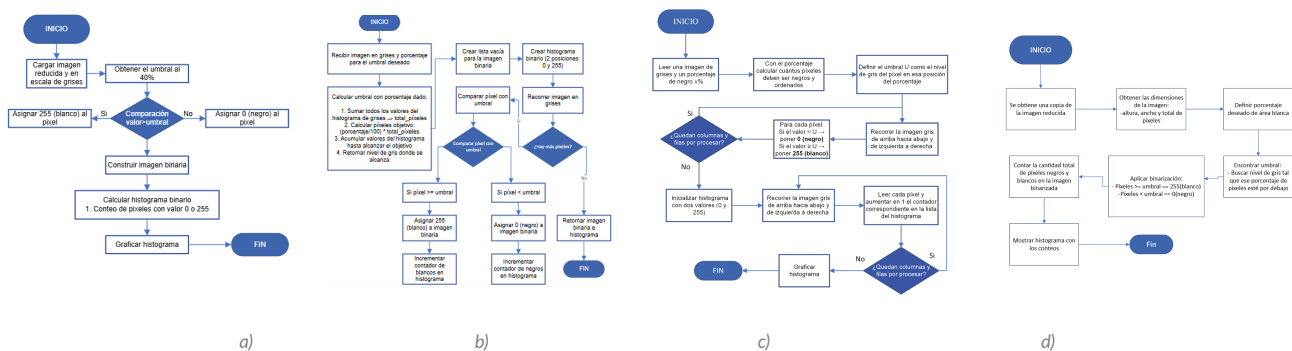


Figura 5. Algoritmos para binarizar una imagen.

Erosión

De acuerdo con la definición *erosión* se desarrollan cuatro diferentes lógicas de programación que se explican a continuación:

Algoritmo a (Fig 6.a). Se examina el elemento estructurante de cada píxel, y se asigna el valor mínimo de ese vecindario al píxel en la imagen de salida, por lo que, se toma una ventana rectangular de 3x7 píxeles, centrada en cada píxel de la imagen.

Algoritmo b (Fig 6.b). Se contempló un kernel personalizable de 3x11. Se utilizó la idea de un píxel como blanco solo si todos los píxeles bajo el kernel son blancos, eliminando así ruido y pequeñas desconexiones en las regiones de interés.

Algoritmo c (Fig 6.c). La erosión reduce las regiones blancas en la imagen binaria. Se utilizó un elemento estructurante de 3x6 píxeles blancos que recorre la imagen y conserva sólo los píxeles donde todos los valores del entorno coinciden. Esto adelgaza los bordes y elimina detalles pequeños en la radiografía dental.

Algoritmo d (Fig 6.d). La erosión es una operación que reduce el tamaño de las áreas blancas en una imagen binaria. Funciona deslizando un pequeño patrón (llamado elemento estructurante) sobre la imagen. Un punto en la imagen resultante será blanco sólo si todos los píxeles cubiertos por ese patrón, al estar centrado en dicho punto, eran blancos en la imagen original.

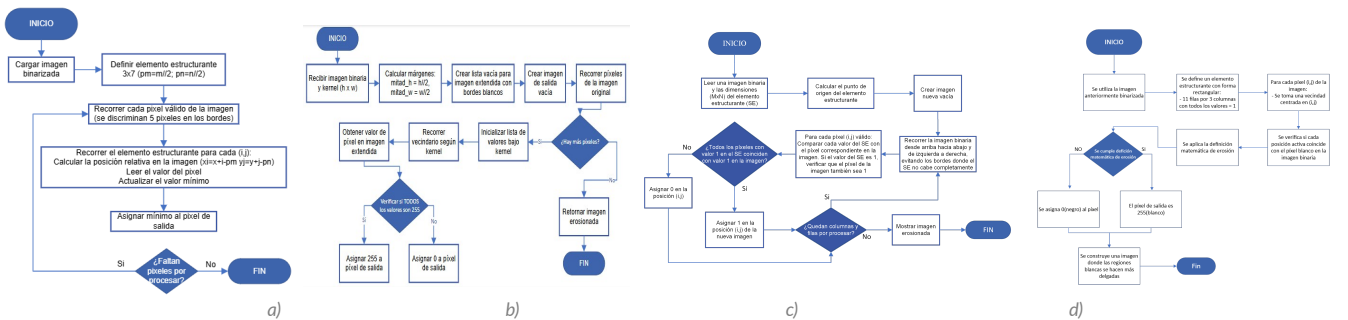


Figura 6. Algoritmos para erosionar una imagen binaria.

Dilatación

De acuerdo con la definición *dilatación* se desarrollan cuatro diferentes lógicas de programación que se explican a continuación:

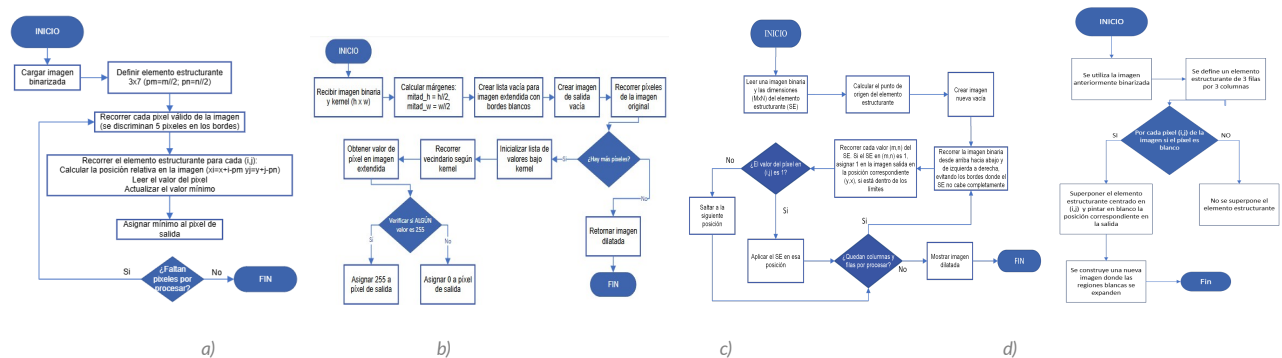


Figura 7. Algoritmos para dilatar una imagen binarizada.

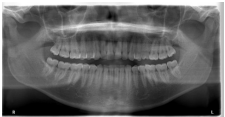
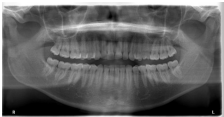
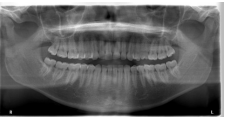
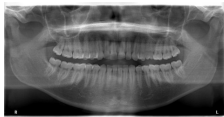
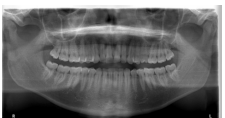
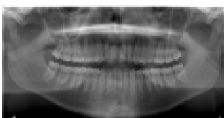
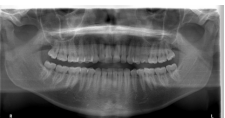
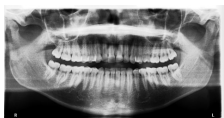
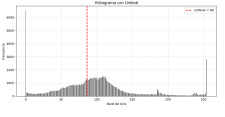
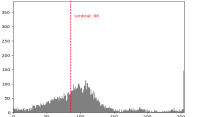
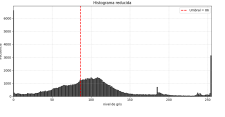
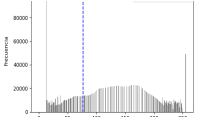




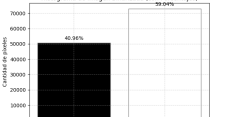
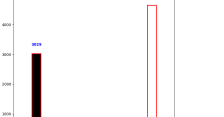

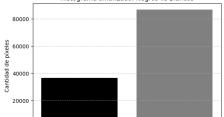








Algoritmo a (Fig 7.a). Contrario a la erosión, se examina el elemento estructurante (3x7) de cada píxel, y se asigna el valor máximo de ese vecindario al píxel en la imagen de salida.

Algoritmo b (Fig 7.b). Se implementó la dilatación como operación dual a la erosión. Se agregaron bordes blancos a la imagen para evitar falsas detecciones en los límites, ya que los bordes negros originales generaban artefactos. Se utilizan kernels asimétricos cuando se requiere mayor expansión en direcciones específicas, en este caso de 3x11.

Algoritmo c (Fig 7.c). La dilatación expande las regiones blancas. Con el mismo elemento estructurante de 3x6, cada píxel blanco en la imagen original propaga su valor sobre su vecindario. Así se engrosan estructuras y se conectan zonas cercanas en la radiografía.

Algoritmo d (Fig 7.d). La dilatación es una operación morfológica que expande las regiones blancas en una imagen binaria. Sirve para rellenar huecos, aumentar el grosor de los objetos y unir componentes cercanos.

Tabla 1. Imágenes resultantes de las operaciones morfológicas realizadas a la radiografía base mediante los algoritmos a), b), c) y d).

Operación morfológica	Algoritmo a	Algoritmo b	Algoritmo c	Algoritmo d
Reducción de resolución				
Escala de grises				
Histograma de Escala de grises				
Binarizado				
Histograma de Binarizado				
Erosión				
Dilatación				

Cálculo de Centroides

De acuerdo con la definición *centroides* se desarrollan cuatro diferentes lógicas de programación que se explican a continuación:

Algoritmo a (Fig 8.a). Se invierte la imagen binaria ya que OpenCV identifica los objetos como regiones de píxeles mayores a 0 y el fondo como píxeles con valor 0, al tener el objeto de interés (región intramaxilar) en píxeles con valor 0, si no se invierte, toma el objeto como fondo, en cambio, al invertirlo, el objeto tiene valor distinto de 0, por lo que OpenCV puede etiquetarlos correctamente como componentes conectados. [22]

Algoritmo b (Fig 8.b). Para calcular centroides, se usó un algoritmo de detección de regiones 8-conectadas, recorriendo la imagen para marcar píxeles no visitados y evitar redundancia. Las coordenadas de cada región se acumularon para obtener su área y centroide (promedio de coordenadas). Finalmente, se ordenaron por tamaño y se seleccionaron las R más grandes.

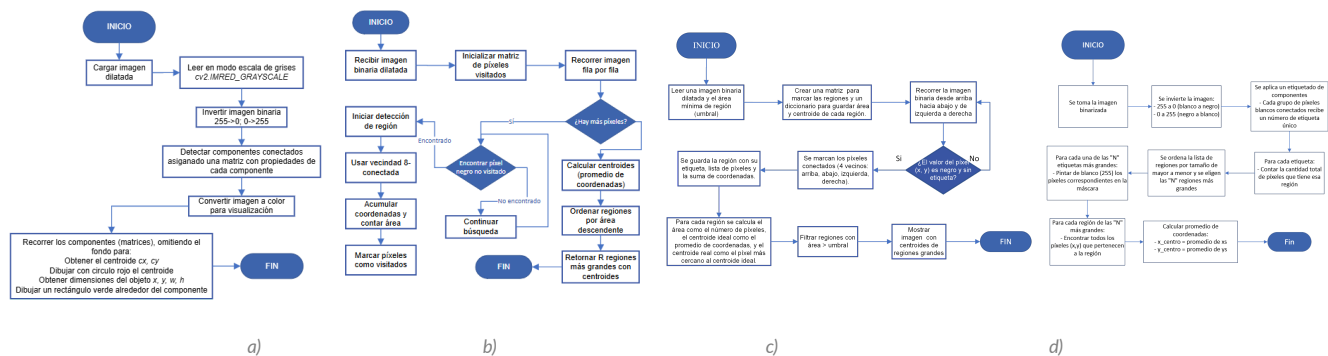


Figura 8. Algoritmos para cálculo de centroides de áreas de interés.

Algoritmo c (Fig 8.c). Para cada región de píxeles negros, se calcula el centroide ideal promediando las posiciones de todos sus píxeles. Luego, se busca el píxel dentro de la misma región que esté más cerca de ese punto promedio, y ese se toma como el centroide real.

Algoritmo d (Fig 8.d). Este código localiza y resalta las regiones más grandes en una imagen binaria. Primero, invierte la imagen (cambia blancos por negros y viceversa) para que los objetos de interés sean blancos. Luego, identifica cada región conectada (conjunto de píxeles blancos adyacentes) y cuenta el número de píxeles en cada una.

Obtención de la región intramaxilar

De acuerdo con la definición *región intramaxilar* se desarrollan cuatro diferentes lógicas de programación que se explican a continuación:

Algoritmo a (Fig 9.a). Se invierte la imagen para que OpenCV pueda detectar adecuadamente los objetos que originalmente tiene píxeles con valor a 0.[22] Se filtran las regiones que tocan los bordes ya que puede clasificar la imagen completa como región de interés. El filtrado por área mínima se hace para eliminar aquellos objetos que cumplen con las características anteriores pero tienen un área pequeña en comparación al área de la región intramaxilar, con estos criterios, se logra detectar correctamente la estructura sin intervención manual.

Algoritmo b (Fig 9.b). Para obtener la región intramaxilar se utilizó floodFill de OpenCV. Primero se reescaló y dilató a un kernel de 5x20 la imagen reducida y las coordenadas del centroide seleccionado. Se aplicó el relleno desde el centroide usando conectividad-4, almacenando el resultado en una máscara. Los píxeles rellenos se marcaron con valor 127 y luego se convirtieron a 0 en la máscara final, mientras el fondo se estableció a 255. Esto permitió aislar la región de interés.

Algoritmo c (Fig 9.c). De entre todas las regiones negras detectadas con área suficiente (>1000 píxeles), se selecciona aquella cuyo centroide esté más cerca del centro geométrico de la imagen. A partir de su etiqueta, se genera una nueva imagen en blanco donde únicamente se conserva esa región en negro.

Algoritmo d (Fig 9.b). Primero, determina el centro geométrico de la imagen. Luego, a través de las regiones más grandes previamente identificadas, calcula el centroide de cada una y la distancia de ese centroide al centro de la imagen. Rastrea la región con la distancia más pequeña. Finalmente, dibuja un círculo en el centroide de la región más cercana y muestra tanto la imagen con el centroide marcado como una máscara que aísla solo esa región.

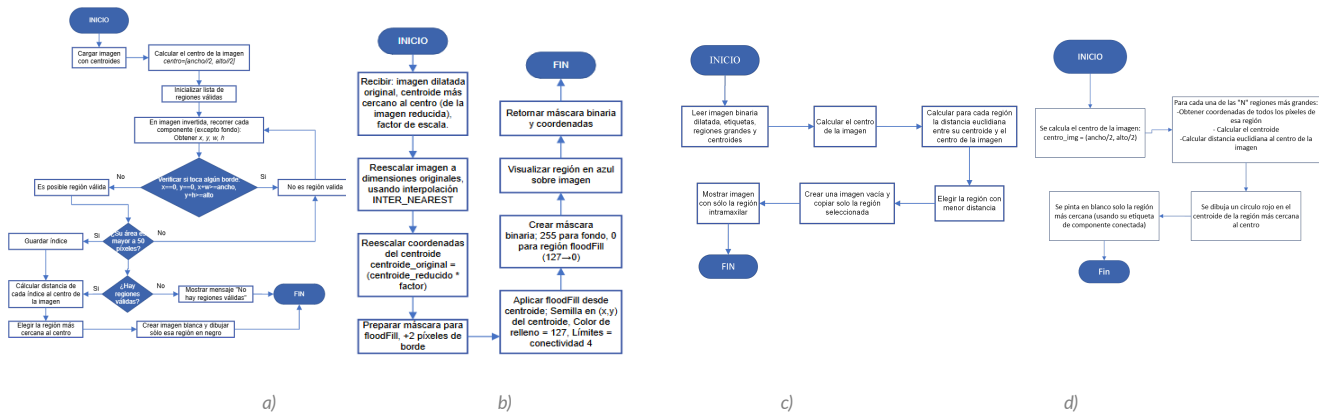


Figura 9. Algoritmos para obtención de la región intramaxilar.

Simetrización

De acuerdo con la definición *simetrización* se desarrollan cuatro diferentes lógicas de programación que se explican a continuación:

Algoritmo a (Fig 10.a). Se reconstruye una región binaria simétrica respecto a un eje vertical, tomando como referencia la mitad más dominante de la región detectada y marcando su eje central. Si el ancho original era impar, se agrega una columna blanca para igualar tamaños y evitar distorsión.

Algoritmo b (Fig 10.b). La simetrización se realizó reflejando la región intramaxilar sobre el eje central vertical de la imagen. Se aplicó una operación AND entre la región original y su reflejo para conservar sólo las áreas simétricas. Finalmente, se filtraron componentes pequeños para eliminar ruido residual, obteniendo una representación bilateralmente simétrica de la anatomía maxilar.

Algoritmo c (Fig 10.c). Este proceso aplica simetría vertical a una región binaria usando como eje el centroide previamente calculado. La mitad con más píxeles negros se toma como base para reflejarla en el lado opuesto.

Algoritmo d (Fig 10.d). Se busca crear una imagen simétrica a partir de una región seleccionada, representando una sonrisa. Primero, encuentra el punto central horizontal de la región. Luego, divide la región en dos mitades: izquierda y derecha.

Comparando el número de píxeles blancos en cada mitad.

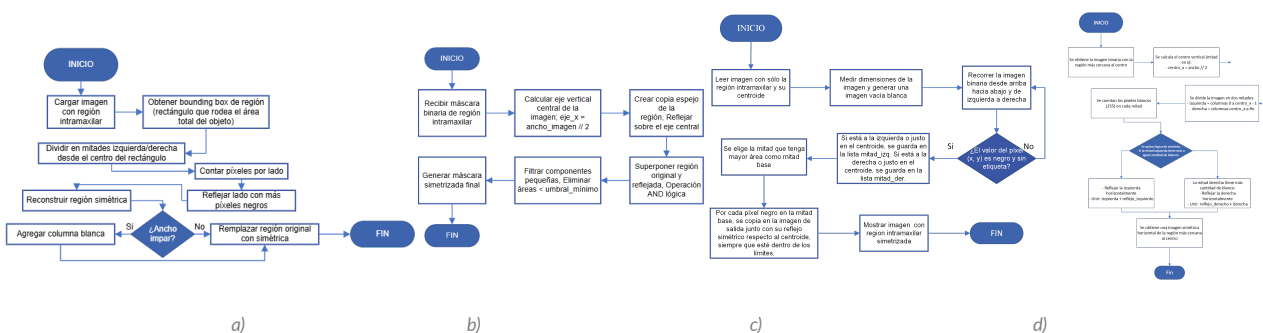


Figura 10. Algoritmos para simetrizar la región intramaxilar.

Obtención de Línea Central

De acuerdo con la definición *línea central* se desarrollan cuatro diferentes lógicas de programación que se explican a continuación:

Algoritmo a (Fig 11.a). Se recorre cada columna dentro del bounding box, en dónde se detectan los extremos superior e inferior de la región negra y se calcula el punto medio vertical y se marca con un punto rojo. Esto representa una línea central de simetría vertical, útil para análisis morfológicos o mediciones dentales.

Algoritmo b (Fig 11.b). La línea central se obtuvo analizando cada columna de la región simétrica. Para cada columna con píxeles negros (región intramaxilar), se calculó el punto medio entre los límites superior e inferior. Estos puntos medios se conectan para formar la línea central que representa el eje de simetría de la sonrisa.

Algoritmo c (Fig 11.c). Este proceso identifica el punto medio de cada columna que contiene píxeles negros en una imagen binaria. Los puntos se conectan con una línea roja para representar visualmente la línea central de la figura.

Algoritmo d (Fig 11.d). Se convierte una imagen en escala de grises a color, luego calcula el promedio vertical de los píxeles blancos en cada columna y dibuja puntos azules en esas posiciones, formando una línea central horizontal.

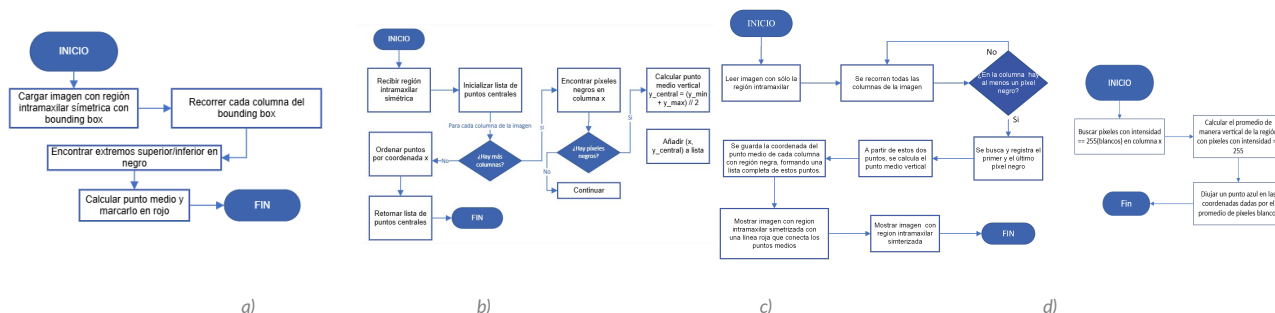


Figura 11. Algoritmos para obtención de línea central de la región intramaxilar.

Obtención de spline

De acuerdo con la definición *spline* se desarrollan cuatro diferentes lógicas de programación que se explican a continuación:

Algoritmo a (Fig 12.a). Se seleccionaron 6 puntos para tener un mayor suavizamiento en la curva spline sin perder información importante de la región.

Algoritmo b (Fig 12.b). Se ajustó un polinomio de grado 6 a los puntos centrales usando interpolación, eliminando duplicados primero. Con 3+ puntos se usó un polinomio de grado controlado; con menos, interpolación lineal.

Algoritmo c (Fig 12.c). Este proceso suaviza la línea central de una región negra usando interpolación polinomial (spline). Se generan 20 puntos equidistantes a lo largo de la curva, con los cuales se genera un polinomio de grado 6.

Algoritmo d (Fig 12.d). Se usa una función spline para trazar una línea roja suave que conecta puntos, mostrando la tendencia central de las áreas blancas.

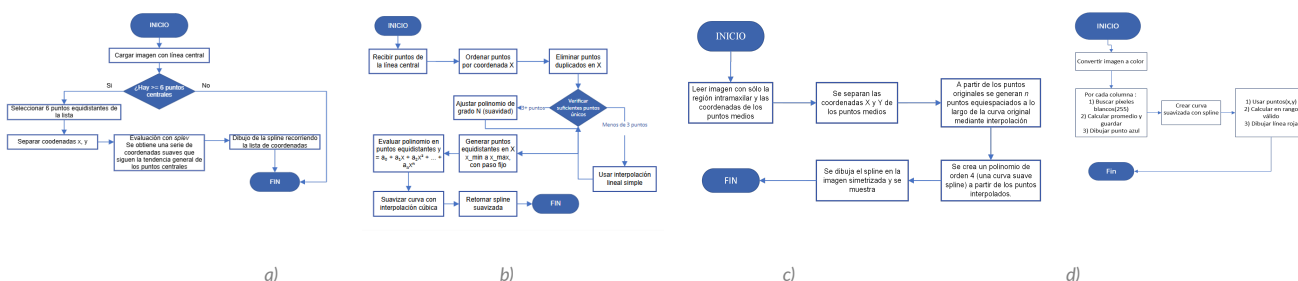


Figura 12. Algoritmos para obtención de spline a partir de la línea central de la región intramaxilar.

Tabla 2. Imágenes resultantes de las operaciones siguientes realizadas a la radiografía base mediante los algoritmos a), b), c) y d).

Operación	Algoritmo a	Algoritmo b	Algoritmo c	Algoritmo d
Cálculo de centroides				
Obtención de región intramaxilar				
Simetrización				
Obtención de línea central				
Obtención de spline				

División en segmentos de la spline

De acuerdo con la definición *división de la spline* se desarrollan cuatro diferentes lógicas de programación que se explican a continuación:

Algoritmo a (Fig 13.a). Se marcan 32 divisiones regulares a lo largo de la spline, como una primera aproximación para análisis dental, para detectar los espacios interdientales correspondientes a cada pieza dental.

Algoritmo b (Fig 13.b). La spline se dividió en 32 segmentos equidistantes mediante interpolación lineal. Primero se calculó la longitud total de la curva sumando las distancias entre sus puntos. Luego, se determinó la longitud objetivo para cada segmento ($\text{longitud_total}/32$). El algoritmo recorrió la spline acumulando distancia e insertando nuevos puntos cuando alcanzaba la longitud objetivo, garantizando una división precisa y uniforme.

Algoritmo c (Fig 13.c). Se proyecta la spline generada en la imagen reducida original y se divide en 32 segmentos para facilitar su análisis.

Algoritmo d (Fig 13.d). Se crea una curva suave (spline) a través de esos puntos y la divide en 32 segmentos equidistantes.

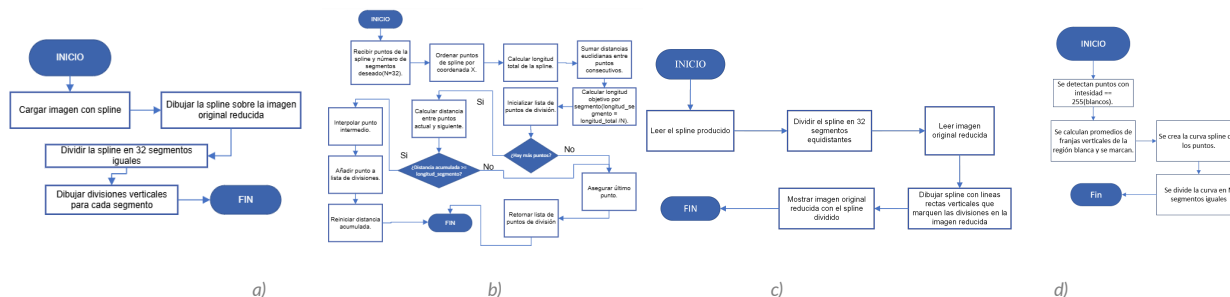


Figura 13. Algoritmos para la división de la spline en segmentos.

Identificación de espacios interdentales

De acuerdo con la definición *espacios interdentales* se desarrollan cuatro diferentes lógicas de programación que se explican a continuación:

Algoritmo a (Fig 14.a). Se utiliza una imagen auxiliar generada desde la combinación entre una imagen complementaria, la cuál su función es resaltar los dientes, y una imagen con el gradiente de sobel que ayuda a detectar los bordes, teniendo como resultado una imagen que mejora la visibilidad de los contornos dentales y refuerza el contraste, adquiriendo una mayor diferencia visual entre los espacios interdentales a las estructuras que los rodean. Al colocar la spline por arriba (arcada superior) o por abajo (arcada inferior) de las coronas, se centra la búsqueda de zonas más brillantes a solo el área de interés, eliminando posibles errores en la detección. Se divide la spline en 17 segmentos definidos por porcentajes adaptados al ancho de cada diente, estos porcentajes se obtuvieron de la tabla I. El filtro de distancia se aplica para conservar solo un punto por cada mínimo 5 píxeles en X, esto evita detecciones múltiples cercanas, reduce el ruido visual y asegura un solo punto por segmento.

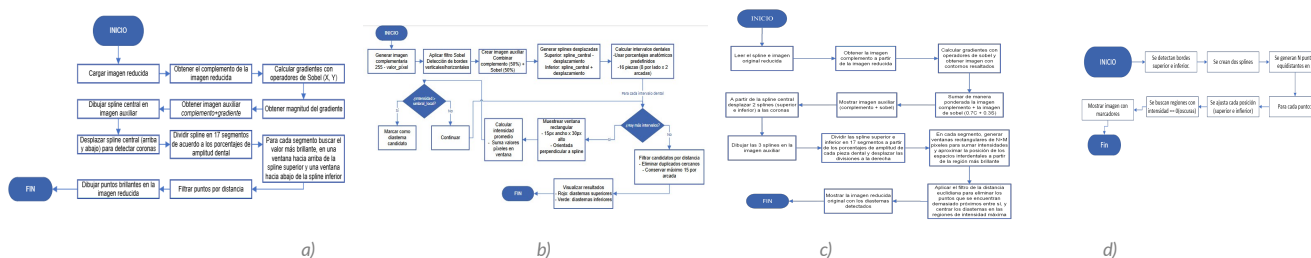


Figura 14. Algoritmos para la identificación de espacios interdentales.

Algoritmo b (Fig 14.b). Primero se generó una imagen auxiliar mezclando el complemento (para resaltar espacios oscuros) y el gradiente Sobel (para enfatizar bordes). Las splines desplazadas superior e inferior se calcularon paralelas a la línea central, manteniendo la anatomía dental. Los intervalos precisos se determinaron usando porcentajes aproximados del ancho de las piezas dentales (Tabla I). Para cada intervalo, se analizó una ventana orientada perpendicularmente a la spline, buscando regiones con alta intensidad (combinación de claridad y bordes). Finalmente, se aplicó un filtro de distancia para eliminar detecciones redundantes, obteniendo diastemas anatómicamente plausibles.

Tabla III. Porcentaje aproximado del ancho de las piezas en una dentadura permanente [17]

Pieza	Arcada	Porcentaje
Tercer Molar	Superior	6.7376 %
	Inferior	7.7762 %
Segundo Molar	Superior	7.1631 %
	Inferior	7.8114 %
Primer Molar	Superior	6.0283 %
	Inferior	6.5095 %
Segundo Premolar	Superior	4.8581 %
	Inferior	4.9613 %
Primer Premolar	Superior	5.2127 %
	Inferior	4.8557 %
Canino	Superior	5.0354 %
	Inferior	4.3983 %
Incisivo Lateral	Superior	5.3900 %
	Inferior	3.8705 %
Incisivo Central	Superior	6.0993 %
	Inferior	3.7297 %

Algoritmo c (Fig 14.c). El algoritmo comienza con la generación de una imagen auxiliar (complemento + sobel) y la proyección en ésta de la spline obtenida previamente, de la cual se derivan dos curvas desplazadas (arcada superior e inferior) para abarcar las zonas donde comúnmente se localizan los espacios interdentes (arriba y abajo de las coronas). A continuación, cada curva se divide en 17 intervalos desiguales, calculados según porcentajes anatómicos promedio (Tabla III) para representar correctamente las 16 piezas dentales. Para evitar que los puntos de detección coincidan con las piezas dentales, se aplica un desplazamiento hacia la derecha en los intervalos, asegurando que la segmentación se sitúe entre los dientes, no sobre ellos. Sobre cada intervalo se proyectan ventanas rectangulares (N×M) que recorren las curvas buscando las regiones más brillantes, asociadas a los posibles diastemas. El punto con mayor intensidad dentro de cada intervalo se selecciona como candidato. Finalmente, se aplica un filtro de distancia euclidiana para eliminar duplicados: si dos puntos están demasiado cerca (menos del 70% del ancho promedio de un diente), se conserva el que se ubica en la zona más brillante. Esto mejora la precisión y asegura una distribución anatómicamente coherente. La distancia euclidiana entre dos vectores a y b se define como:

$$d(a, b) = \|a - b\| \rightarrow d(a, b) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \quad (14)[17]$$

Algoritmo d (Fig 14.d) Se toma la imagen anteriormente reducida en escala de grises y a partir de una máscara, identifica los límites superior e inferior de las regiones blancas. Luego, traza dos curvas suaves (splines) que siguen estos límites. En puntos específicos a lo largo de estas curvas, el código verifica si hay una región oscura cercana. Si se detecta una región oscura, dibuja una línea vertical azul sobre el spline superior y una línea vertical verde debajo del spline inferior. Finalmente, muestra la imagen original con estas líneas superpuestas, indicando visualmente dónde las áreas blancas de la máscara colindan con zonas oscuras de la imagen de entrada.


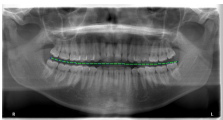
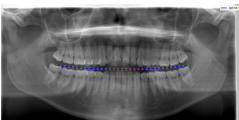
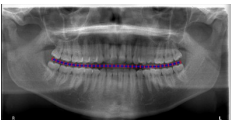
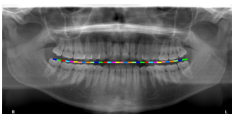

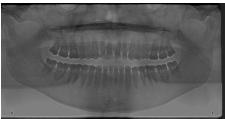

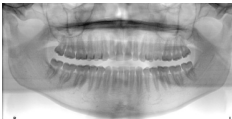
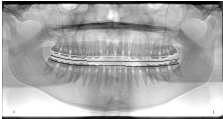
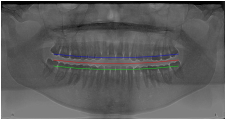
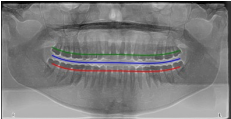
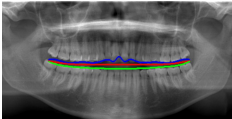
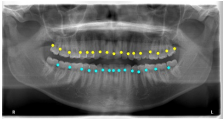
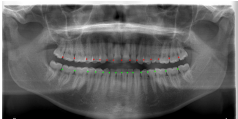
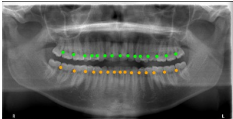
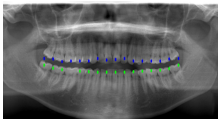
Los resultados de cada una de las partes del algoritmo se encuentran en las tablas I, II y IV de igual forma, se hizo el análisis de cada algoritmo y se concluyó que, aunque las lógicas de programación son diferentes en cada caso, el objetivo de lograr la identificación de espacios interdentes en un tiempo menor al que la inteligencia artificial lo logra se completo, ya que con estos códigos se tiene un promedio de procesamiento de máximo 15 segundos. El compilado de los códigos se encuentra en:  **Compilación de códigos a), b), c) y d)**

Tabla 4. *Imágenes resultantes de las operaciones finales realizadas a la radiografía base mediante los algoritmos a), b), c) y d).*

Operación	Algoritmo a	Algoritmo b	Algoritmo c	Algoritmo d
División de spline				
	<i>Imagen auxiliar</i>	<i>Imagen auxiliar</i>	<i>Imagen auxiliar</i>	<i>Imagen auxiliar</i>
				
Identificación de diastemas	<i>Imagen con splines</i>	<i>Imagen con splines</i>	<i>Imagen con splines</i>	<i>Imagen con splines</i>
				
	<i>Imagen con espacios interdentes</i>	<i>Imagen con espacios interdentes</i>	<i>Imagen con espacios interdentes</i>	<i>Imagen con espacios interdentes</i>
				

Conclusiones

El desarrollo del algoritmo demuestra que es posible implementar herramientas computacionales capaces de adaptarse a las variaciones anatómicas y técnicas presentes en este tipo de imágenes médicas. A través de la combinación de operaciones morfológicas, procesamiento, análisis de centroides y el ajuste de splines, fue posible aislar de forma eficiente la región intramaxilar, generar líneas centrales suavizadas y localizar con precisión los espacios interdentes relevantes para el diagnóstico. Cada una de las versiones propuestas del algoritmo tiene fortalezas particulares en etapas específicas del procesamiento, permitiendo comparar enfoques alternativos para cada paso. La incorporación del análisis por segmentos, junto con el uso del complemento de imagen y operadores de Sobel, permitió aumentar significativamente la visibilidad y precisión en la delimitación de los espacios interdentes, incluso en condiciones de bajo contraste o ruido estructural. Los resultados obtenidos en las imágenes de prueba sugieren que este enfoque puede constituirse como una herramienta de apoyo clínico. Asimismo, el identificar los espacios interdentes en una radiografía panorámica dental sienta las bases para futuras mejoras que incorporan técnicas de aprendizaje automático o análisis volumétrico, lo cual podría ampliar su aplicabilidad en contextos odontológicos reales, como el análisis morfológico de las piezas dentales para analizar propiedades de interés o alguna malformación que a simple vista no es evidente.

Agradecimientos

Por parte del equipo de investigación se extiende un cordial agradecimiento a todas aquellas instituciones y personalidades que contribuyeron a la realización de este trabajo, se agradece a la Universidad de Guanajuato y al programa Verano de la Ciencia por la oportunidad de ser parte de proyectos de investigación de alto alcance, al Centro de Investigaciones en Óptica por facilitar las instalaciones y recursos para poder desarrollar los productos obtenidos a lo largo de la estancia, a la Facultad de Odontología con la colaboración de la Dra. Miriam Rocha de la Universidad de La Salle Bajío por el contacto con la clínica, facilitando imágenes de radiografías panorámicas dentales de diferentes pacientes para poder probar y mejorar los algoritmos creados, y por último a las instancias gubernamentales SECIHTI, al Gobierno Estatal de Guanajuato y todas aquellas instancias cuyo objetivo es el desarrollo de ciencia y tecnología en el país y que directa o indirectamente su trabajo impacta en la realización de proyectos de investigación como este.

Bibliografía/Referencias

- [1] Jaiswal, P., & Bhirud, S. (2023). A cropping algorithm for automatically extracting regions of interest from panoramic radiographs based on maxilla and mandible parts. *International Journal of Information Technology*, 15(12), 3631–3641. <https://doi.org/10.1007/s41870-023-01406-4>
- [2] Silva, G., Oliveira, L., & Pithon, M. (2018). Automatic segmenting teeth in X-ray images: Trends, a novel data set, benchmarking and future perspectives. *Expert Systems with Applications*, 107, 15-31.
- [3] Mello Román, J. C., Fretes, V. R., Adorno, C. G., Gariba Silva, R., Vázquez Noguera, J. L., Legal-Ayala, H., Mello-Román, J. D., Escobar Torres, R. D., & Facon, J. (2021). Panoramic Dental Radiography Image Enhancement Using Multiscale Mathematical Morphology. *Sensors*, 21(9), 3110. <https://doi.org/10.3390/s21093110>
- [4] González, R. C., & Woods, R. E. (2018). Digital image processing (4th ed.). Pearson.
- [5] Naik, A., Tikhe, S., Bhide, S., Kaliyamurthi K. P., & Saravanan T. (2016). "Algorithm to Detect Fracture from OPG Images Using Texture Analysis," 2016 IEEE 6th International Conference on Advanced Computing (IACC), Bhimavaram, India, 382-385, <https://doi.org/10.1109/IACC.2016.77>.
- [6] Naik, A., Tikhe, S., Bhide, S., Kaliyamurthi, K. P., & Saravanan, T. (2016). Automatic Segmentation of Lower Jaw and Mandibular Bone in Digital Dental Panoramic Radiographs. *Indian Journal Of Science And Technology*, 9(21), 1-6. <https://indjst.org/articles/automatic-segmentation-of-lower-jaw-and-mandibular-bone-in-digital-dental-panoramic-radiographs>
- [7] Bovik, A. C. (2009). Basic gray level image processing. In A. Bovik (Ed.), *The essential guide to image processing* (pp. 43-68). Academic Press.
- [8] Bovik, A. C. (Ed.). (2005). *Handbook of image and video processing* (2nd ed.). Academic Press.
- [9] Poynton, C. (2003). *Digital video and HDTV: Algorithms and interfaces*. Morgan Kaufmann.
- [10] International Telecommunication Union. (2011). Recommendation ITU-R BT.601-7: Studio encoding parameters of digital television for standard 4:3 and wide-screen 16:9 aspect ratios. International Telecommunication Union. https://www.itu.int/dms_pubrec/itu-r/rec/bt/r-rec-bt.601-7-201103-i!!pdf-e.pdf
- [11] Pajares, M. G., & de la Cruz, G. J. M. (2008). *Visión por computador. Imágenes Digitales y Aplicaciones*. (2a Ed). RA-MA S.A.
- [12] Kaur, R., Sandhu, R. S., Gera, A., & Kaur, T. (2017). Edge detection in digital panoramic dental radiograph using improved morphological gradient and MATLAB. *International Conference On Smart Technologies For Smart Nation (SmartTechCon)*, 793-797. <https://doi.org/10.1109/smarttechcon.2017.8358481>
- [13] Montes, T. O., & Castillo, P. A. C. (1996). La morfología matemática en el tratamiento digital de imágenes. *Ensayos: Revista de la Facultad de Educación de Albacete*, (11), 241-256.
- [14] Amer, Y. Y., & Aqel, M. J. (2015). An efficient segmentation algorithm for panoramic dental images. *Procedia Computer Science*, 65, 718-725.

- [15] Feral, V. J. (1994). ANÁLISIS DE IMÁGENES y MORFOLOGÍA MATEMÁTICA: Descripción de herramientas básicas. Centro de investigación y desarrollo tecnológico en electroquímica del estado de Querétaro.
- [16] Mallya, S., & Lam, E. (2018). White and Pharoah's oral radiology: principles and interpretation. Elsevier Health Sciences.
- [17] García Alcalá, L. M. (2020). Segmentación de radiografías panorámicas dentales mediante técnicas de procesamiento digital de imágenes [Tesis de Licenciatura]. Instituto Tecnológico de Estudios Superiores de Zamora.
- [18] Loy, G., & Eklundh, J. (2006). Detecting Symmetry and Symmetric Constellations of Features. *SpringerNature*. LNCS 3952, pp. 508–521. https://doi.org/10.1007/11744047_39
- [19] Cuevas, F., Rocha Navarro, M., Garcia Salido, M., & Rodríguez Villegas, M. (2023). Metrology and Digital Image Processing in Dentistry. IntechOpen. <https://doi.org/10.5772/intechopen.108375>.
- [20] Abrahams, R., & Kamath, G. (2014). Midline diastema and its aetiology—a review. *Dental update*, 41(5), 457-464.
- [21] Martin, M. (2002). Técnicas clásicas de segmentación de imagen.
- [22] Bradski, G. (2000). The OpenCV Library. OpenCV.org. <https://docs.opencv.org/>