

UNIVERSIDAD DISTRITAL FRANCISCO JOSÉ DE CALDAS

MONOGRAFIA

Análisis del blanqueamiento coralino en fotografías subacuáticas con redes neuronales artificiales

Autor:

Carol Zuleimy Fernández Rodríguez

Director:

M. Sc. Wilson Ricardo López
Sánchez



UNIVERSIDAD DISTRITAL
FRANCISCO JOSÉ DE CALDAS

Grupo de Investigación
GITEM

Ingeniería Electrónica

9 de mayo de 2024

Índice general

1.	Introducción	2
2.	Justificación	3
3.	Objetivos	5
	3.0.1. Objetivo general	5
	3.0.2. Objetivos específicos	5
4.	Estado del arte	6
	4.1. Quantifying the degree of coral bleaching using digital photographic technique	6
	4.2. Prediction of coral bleaching in the Florida Keys using remotely sensed data	7
	4.3. Comparison of three digital image analysis techniques for assessment of coral cover and bleaching	8
5.	Marco teórico	10
	5.1. Conceptos básicos de redes neuronales artificiales	10
	5.1.1. Elementos y funciones de una red neuronal	10
	5.1.2. Métodos de aprendizaje	12
	5.1.3. Topologías de redes neuronales	13
	5.2. Aplicaciones de Redes Neuronales Artificiales para el análisis de imágenes	15
	5.2.1. Coral species identification with texture or structure images using a two-level classifier based on Convolutional Neural Networks	15
	5.2.2. Recognition of Coronary Arterial Stenosis Using Neural Network on DSA System	16
	5.2.3. Three-dimensional medical image analysis of the heart by the revised GMDH-type neural network self-selecting optimum neural network architecture	17
	5.3. Técnicas de procesamiento digital de imágenes para la detección de forma y color	18
	5.3.1. Segmentación por color	18
	5.3.2. Filtros morfológicos	19
6.	Metodología	21
7.	Desarrollo	23
	7.1. Base de datos seleccionada	23
	7.1.1. Proceso de segmentación manual	23
	7.2. Exploración y selección de modelos de detección y segmentación de imágenes digitales con redes neuronales	24
	7.2.1. Detección de imágenes digitales con redes neuronales	24
	7.2.2. Segmentación de imágenes digitales con redes neuronales	30
	7.2.3. Selección de los modelos	34
	7.2.4. Primeras pruebas	35
	7.3. Parametrización de los modelos	39
	7.3.1. text prompt	39
	7.3.2. Umbral de detección para GroundingDino	43

7.3.3.	Variantes y Checksums de SAM	43
7.4.	Detección de color (fase 3)	43
7.4.1.	Selección del algoritmo de discriminación de color	44
7.4.2.	Luminancia	44
7.4.3.	K-means	47
7.4.4.	Parametrización k-means	48
7.4.5.	Descripción detallada	49
7.4.6.	Centroides aleatorios	49
7.4.7.	Centroides basados en Kmeans++	50
8.	Resultados	52
9.	Conclusiones	59
10.	Anexos	61

Índice de figuras

1.	Fotografía subacuática de un coral con blanqueamiento (abajo) y un coral sano (arriba). Tomado de “Climate Change, Human Impact, and the Resilience of Coral Reefs” [19]	4
2.	Variaciones en la exposición de la fotografía de una pieza de coral con blanqueamiento [12]	6
3.	Estimación del blanqueamiento con relacion a las variaciones de exposición [12]	7
4.	Región de estudio y localización de los sensores remotos en esa área [9]	8
5.	Estimación del porcentaje de coral sano y coral blanqueado para los tres métodos estudiados [22]	9
6.	Modelo básico de una neurona [34]	10
7.	Función sigmoide	11
8.	Función tangente hiperbólica	12
9.	Función RELU	12
10.	Topología red neuronal feedforward [38]	13
11.	Topología red neuronal recurrente [38]	14
12.	Arquitectura de la red neuronal convolucional [11]	14
13.	Capa convolucional [6]	15
14.	Diferencia entre una imagen coralina de textura y estructura. Izquierda, imagen de textura; derecha, imagen de estructura [15]	16
15.	Resultados del reconocimiento de estenosis [39]	17
16.	Análisis de imágenes tomográficas computarizadas de rayos X del corazón: Izquierda, imagen original; derecha, imagen de salida de la red neuronal [24]	18
17.	Ejemplo de segmentación por color: tuercas de colores [33]	18
18.	Ejemplo de segmentación por color: dulces de colores [33]	19
19.	Extracción de autobuses en una imagen con filtros morfológicos [36]	20
20.	Arquitectura propuesta para la solución del problema	22
21.	EfficientDet: Algunos objetos que detecta el modelo [40]	25
22.	OWL-ViT: Detección de objetos en mundo abierto con Vision Transformers [31]	26
23.	Modelo Grounding DINO [27]	27
24.	Ground DINO: Explicación de las entradas y salidas [27]	28
25.	MobileNet: Uso en entornos urbanos [18]	29
26.	Detección de animales marinos en tiempo real utilizando redes de aprendizaje profundo basadas en YOLO en el ecosistema de arrecife de coral [43]	30
27.	Modelo Segment Anything (SAM) [23]	30
28.	Ejemplo de segmentación automática usando SAM, con imágenes del conjunto de datos SA-1B [23]	31
29.	SEEM admite tareas de segmentación genéricas de forma abierta cuando no se proporciona ningún mensaje. También permite el uso de indicaciones visuales, textuales y de regiones de referencia.[44]	32

30.	SegGPT es capaz de segmentar todo en contexto con un solo modelo, que utiliza ejemplos en contexto para indicar diferentes tareas. Para cada muestra, el cuadro naranja a la izquierda muestra la imagen de ejemplo/solicitud y su máscara correspondiente, mientras que el cuadro azul a la derecha muestra la imagen de entrada y la salida de máscara resultante. La máscara representa la región brillante adjunta a la imagen. El título de cada muestra (en el cuadro amarillo) es solo para explicación.[41]	33
31.	Clasificación del desastre del terremoto con diferentes fondos circundantes de objetos terrestres, (a) área de color uniforme, (b) distribución de color compleja y área de contraste grande, (c) distribución compleja de objetos terrestres circundantes. El marco de la línea roja es el resultado dibujado a mano mediante interpretación visual, y las áreas violeta y amarilla son los resultados de clasificación de los desastres sísmicos obtenidos por el modelo DMCNN.[42]	34
32.	Ejemplo segmentación completa sin semántica y con parámetros por defecto	36
33.	Ejemplo segmentación completa sin semántica y con parámetros por defecto [4]	37
34.	Detección de objetos sin parametrizar, con el prompt "coral"	39
35.	Máscaras resultantes de SAM con parametrización incorrecta	39
36.	Coral con 1 sección de coral blanqueado y poco fondo	40
37.	Coral en su mayoría sano y 6 regiones de diferentes tamaños blanqueadas	40
38.	Varios corales individuales, muchos de los cuales están blanqueados	41
39.	Dos corales diferentes pero blanqueados, fondo marino apreciable	41
40.	Casi toda la imagen tiene coral blanqueado	41
41.	Imagen completa sin coral blanqueado	41
42.	Coral rojo con 2 regiones idendificables de coral blanqueado	42
43.	Coral blanqueado en primer plano, junto con otras partes de coral sano y bastante fondo marino adicional	42
44.	Coral blanqueado en primer plano, junto con piedras y bastante fondo marino adicional	42
45.	Coral sano y Letras blancas	42
46.	Coral blanqueado uniforme junto con coral sano	43
47.	Coral mixto, partes blanqueadas y partes sanas	43
48.	Ejemplo de corte por luminancia, con diferentes valores para el mismo coral ya segmentado	46
49.	Ejemplo K-means clustering para una imagen de un pájaro colorido	47
50.	Ejemplo de clusterización de una imagen con el algoritmo k-means clustering	48
51.	Ejemplo de cuantización de color, por los diferentes métodos de inicialización para k-means	51

Índice de cuadros

1.	Tabla con las imágenes originales, la segmentación manual y la resultante del modelo .	53
2.	Tabla de resultados y error del modelo	54
3.	Tabla de segmentación hecha por el experto, la resultante del modelo y las diferencias en las imágenes pixel a pixel	56
4.	Tabla con ejemplos de imágenes originales, la cuantización en fase 3 por luminancia y la cuantización en fase 3 por k-means	58
5.	Tabla con las imágenes originales, la segmentación manual y la resultante del modelo .	68
6.	Tabla de segmentación hecha por el experto, la resultante del modelo y las diferencias en las imágenes pixel a pixel	76
7.	Tabla de resultados y error del modelo	77

1. Introducción

El blanqueamiento coralino, fenómeno que afecta a los arrecifes de coral en todo el mundo, es un indicador crucial del estado de salud de los ecosistemas marinos. La capacidad de monitorear y evaluar el blanqueamiento de manera precisa y eficiente es fundamental para comprender su impacto y tomar medidas de conservación efectivas. En este contexto, el uso de tecnologías de visión computacional y redes neuronales artificiales ofrece una prometedora vía para analizar imágenes subacuáticas y cuantificar el blanqueamiento coralino de manera automatizada y precisa.

El presente trabajo aborda esta problemática mediante el desarrollo de una metodología basada en redes neuronales artificiales, diseñada para analizar fotografías subacuáticas y determinar el porcentaje de blanqueamiento presente en los corales. La metodología propuesta consta de tres fases distintas, cada una dirigida a abordar una etapa específica del proceso de análisis.

En la primera fase, se emplea el modelo de detección de objetos Grounding Dino para identificar la presencia de corales en las imágenes. Posteriormente, en la segunda fase, se utiliza el modelo de segmentación Segment Anything para aislar el coral del resto de la imagen, facilitando así la identificación de las áreas afectadas por el blanqueamiento. Finalmente, en la tercera fase, se implementa el algoritmo de clustering K-means para cuantificar los colores presentes en el coral segmentado y calcular el porcentaje de blanqueamiento.

Para validar la metodología desarrollada, se utilizó un banco de imágenes compuesto por 40 fotografías de corales de diversas especies. Cada imagen fue sometida a un proceso de segmentación manual con la ayuda de un biólogo marino experto en corales, con el propósito de establecer una referencia precisa para la evaluación del desempeño del modelo propuesto. Los resultados obtenidos revelaron una alta eficacia en la detección y segmentación de los corales, así como una adecuada identificación de las áreas afectadas por el blanqueamiento en la mayoría de los casos. No obstante, se identificaron desafíos específicos relacionados con la fase de cuantización de colores, especialmente en situaciones donde la iluminación y las características particulares del coral afectaron la precisión del algoritmo.

El presente estudio representa un paso significativo hacia el desarrollo de herramientas automatizadas para el monitoreo del blanqueamiento coralino, ofreciendo nuevas perspectivas para la investigación y conservación de los ecosistemas marinos. Además, identifica áreas de mejora y futuras líneas de investigación orientadas a perfeccionar la metodología propuesta y abordar los desafíos identificados.

2. Justificación

Los arrecifes de coral representan los ecosistemas marinos más diversos y vitales del planeta, albergando entre un cuarto y un tercio de todas las especies marinas conocidas [32].

Además de su extraordinaria biodiversidad, los arrecifes brindan una serie de servicios ecosistémicos fundamentales, tales como evitar la erosión costera, ya que con la morfología que poseen sirven como rompeolas natural y esta protección se hace más evidente durante las tormentas y huracanes, evitando a la vez, el riesgo de inundación [17], por otra parte, son un importante sustento para la pesca, el turismo y la medicina [37]. Sin embargo, estos ecosistemas enfrentan una creciente amenaza debido al cambio climático y otras actividades humanas. El aumento de las temperaturas oceánicas, atribuido en gran medida a la acumulación de gases de efecto invernadero, ha generado un estrés térmico significativo en los arrecifes de coral, desencadenando eventos masivos de blanqueamiento coralino [26]. Este fenómeno ocurre cuando los corales pierden sus algas simbióticas, lo que resulta en su debilitamiento y muerte. Como consecuencia, un tercio de las especies de coral están actualmente en peligro de extinción [37].

Debido a esta problemática, combinada con otras perturbaciones en el medio ambiente por parte del hombre, los corales se encuentran en gran amenaza. A medida que las temperaturas globales continúan aumentando, se espera que la amenaza para los arrecifes de coral aumente significativamente. Para 2011, el 19 % de los arrecifes se había perdido y el 75 % estaba amenazado [16], la mayoría de los estudios que examinan la respuesta de los corales a los modelos globales de circulación atmosférica-oceánica predicen que dentro de los próximos 80 años pocos arrecifes de coral sobrevivirán en los océanos tropicales [37].

Ante esta preocupante situación, es imperativo comprender y monitorear de manera efectiva el blanqueamiento coralino para desarrollar estrategias de conservación y mitigación. En este contexto, el análisis de imágenes subacuáticas se ha vuelto una herramienta invaluable. La fotografía digital submarina permite documentar los cambios en los arrecifes de coral y ha habilitado el desarrollo de diversas técnicas de análisis, incluida la identificación del blanqueamiento. En la figura 1 se puede observar un ejemplo de un coral que presenta blanqueamiento, junto con uno que no lo tiene, este tipo de fotografías son las que se capturan para su estudio.



FIGURA 1: Fotografía subacuática de un coral con blanqueamiento (abajo) y un coral sano (arriba). Tomado de "Climate Change, Human Impact, and the Resilience of Coral Reefs" [19]

En particular, las redes neuronales artificiales (RNA) han surgido como una herramienta prometedora para el análisis de imágenes de arrecifes de coral. Inspiradas en el funcionamiento del sistema nervioso biológico, las RNA son capaces de aprender y reconocer patrones complejos en los datos de entrada [21]. Esta capacidad las hace especialmente adecuadas para tareas de procesamiento de imágenes, como la identificación de áreas afectadas por el blanqueamiento coralino.

El presente proyecto se propone emplear RNA para analizar imágenes subacuáticas y cuantificar el blanqueamiento coralino de manera automatizada y precisa. A través de una metodología de tres fases, que incluye la detección del coral, la segmentación de las áreas afectadas y la cuantificación del blanqueamiento, se busca desarrollar una herramienta efectiva para monitorear y evaluar el estado de los arrecifes de coral. La utilización de RNA en este contexto presenta numerosas ventajas, incluida su capacidad para trabajar con grandes volúmenes de datos y su potencial para reconocer patrones complejos en las imágenes. Si bien existen otros enfoques para el análisis de imágenes de arrecifes de coral, las RNA ofrecen una combinación única de precisión y eficiencia que las hace especialmente atractivas para este tipo de aplicaciones [5].

En conclusión, el presente estudio busca contribuir al entendimiento y la conservación de los arrecifes de coral mediante el desarrollo de una metodología innovadora basada en RNA para el análisis de imágenes subacuáticas. Se espera que los resultados de esta investigación no solo mejoren nuestra comprensión del blanqueamiento coralino, sino que también proporcionen herramientas efectivas para su monitoreo y gestión en el futuro.

3. Objetivos

3.0.1. Objetivo general

- Diseñar un modelo de redes neuronales artificiales, que permita identificar el blanqueamiento de arrecifes de coral a partir de imágenes.

3.0.2. Objetivos específicos

- Explorar y escoger una arquitectura de redes neuronales que realice adecuadamente la tarea de detección y segmentación de imágenes y evaluar su funcionamiento para segmentar los corales de la imagen.
- Parametrizar la arquitectura seleccionada para afinar su desempeño de detección y segmentación de corales, en aras de obtener un mejor resultado en esta tarea específica.
- Implementar una fase adicional de identificación de color para identificar el blanqueamiento del coral.
- Evaluar el rendimiento de la arquitectura frente a la detección del porcentaje del blanqueamiento de corales en imágenes.

4. Estado del arte

4.1. Quantifying the degree of coral bleaching using digital photographic technique

Este estudio consistió en el desarrollo y aplicación de una técnica fotográfica digital para cuantificar el grado de blanqueamiento coralino. El método utilizado para calcular la cobertura de blanqueamiento en los corales en el estudio fue basado en análisis de imágenes digitales.

Los autores proponen un método para cuantificar el blanqueamiento coralino mediante técnicas fotográficas digitales. Esto implica tomar fotografías de los corales antes, durante y después del evento de blanqueamiento. Luego, utilizan software de análisis de imágenes para procesar las fotografías y cuantificar el grado de blanqueamiento.

El proceso de análisis implica varios pasos. Primero, alteran las fotografías capturadas en cuanto al nivel de brillo, le colocaron filtros de diferentes colores para que queden de diferente tonalidad, con variaciones en el contraste y incluso variaciones en la intensidad del color. Se seleccionan áreas de interés en las fotografías que representen el coral y luego, se utiliza un software para calcular la cobertura de blanqueamiento en estas áreas. Este cálculo se realiza midiendo la proporción de píxeles blancos en relación con el total de píxeles en el área seleccionada.

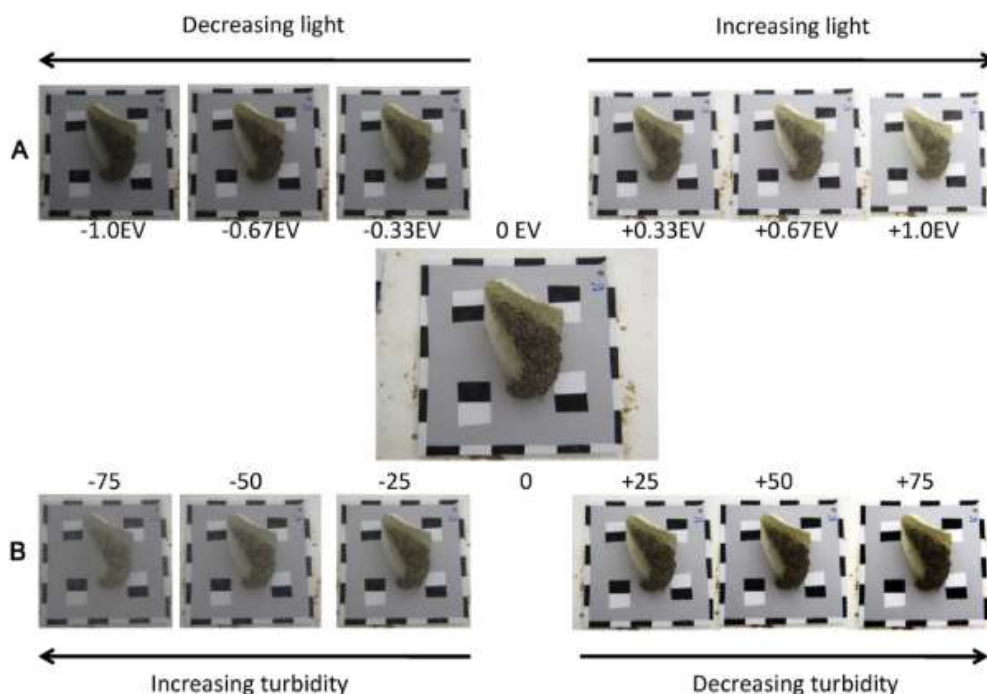


FIGURA 2: Variaciones en la exposición de la fotografía de una pieza de coral con blanqueamiento [12]

Finalmente validan su método comparando los resultados obtenidos mediante fotografías con mediciones in situ realizadas por buzos. Este proceso de validación garantiza que el método propuesto sea preciso y confiable. Los resultados muestran que el método propuesto es efectivo para cuantificar el grado de blanqueamiento coralino. Además, demuestran una alta correlación entre las mediciones obtenidas mediante fotografías y las mediciones in situ, lo que respalda la validez del método.

Presentando un método confiable para cuantificar el blanqueamiento coralino utilizando técnicas fotográficas digitales [12].

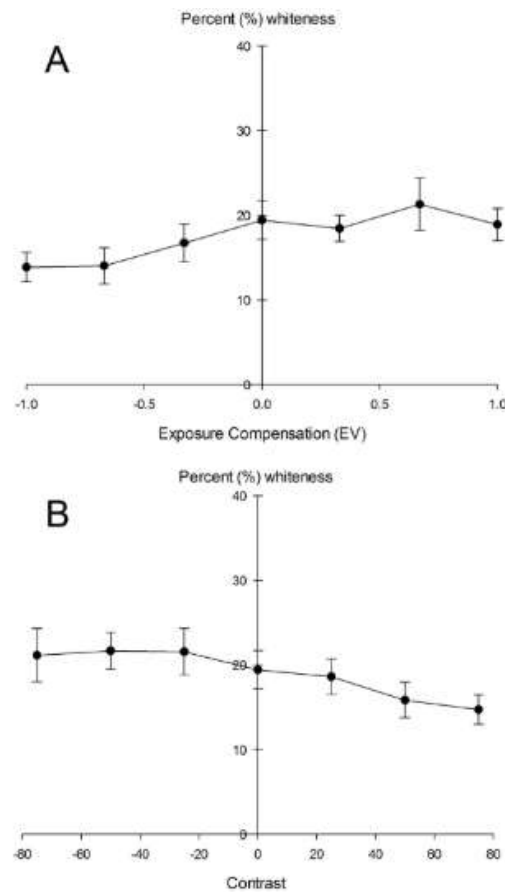


FIGURA 3: Estimación del blanqueamiento con relacion a las variaciones de exposición [12]

4.2. Prediction of coral bleaching in the Florida Keys using remotely sensed data

Este estudio se centra en la predicción del blanqueamiento de los corales en los Cayos de Florida mediante el uso de datos obtenidos de sensores remotos.

Utiliza datos de sensores remotos para monitorear condiciones ambientales clave, como la temperatura del agua y la radiación solar, que son factores importantes en el blanqueamiento de los corales. Se emplea una combinación de imágenes de satélite y modelos estadísticos para predecir los eventos de blanqueamiento con anticipación.

Los investigadores recopilaron datos de temperatura superficial del mar y otros parámetros ambientales durante varios años y los compararon con eventos conocidos de blanqueamiento de corales en la región. Luego, utilizaron modelos estadísticos para identificar patrones y correlaciones entre las condiciones ambientales y los eventos de blanqueamiento.

Los modelos predictivos fueron validados utilizando datos observacionales de eventos de blanqueamiento coralino pasados en la región de estudio. Esto implicó comparar las predicciones generadas por los modelos con los eventos reales de blanqueamiento observados, evaluando la precisión y la

fiabilidad de las predicciones.

Los resultados del estudio muestran que es posible predecir con cierta precisión los eventos de blanqueamiento de corales utilizando datos de sensores remotos. Este modelo predijo condiciones de "blanqueamiento severo" sin blanqueamiento con un 69 y un 57 % de éxito en la clasificación, respectivamente. Esto proporciona una herramienta importante para la gestión de los arrecifes de coral, ya que permite a los administradores tomar medidas preventivas para proteger los arrecifes antes de que ocurran los eventos de blanqueamiento [9].

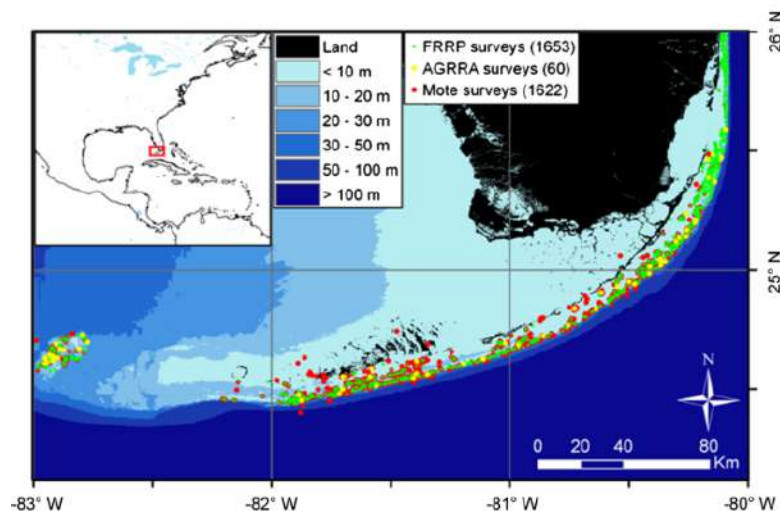


FIGURA 4: Región de estudio y localización de los sensores remotos en esa área [9]

4.3. Comparison of three digital image analysis techniques for assessment of coral cover and bleaching

En este estudio compararon tres métodos de análisis de imágenes digitales para evaluar la cobertura coralina y el blanqueamiento de corales en arrecifes. Los métodos analizados fueron el método de cuadrícula de puntos, el análisis de umbral de color y la clasificación visual.

En el método de cuadrícula de puntos, los investigadores colocaron una cuadrícula sobre las imágenes y contaron los puntos que caían sobre corales vivos y blanqueados para calcular la cobertura coralina y la extensión del blanqueamiento.

Por otro lado, el análisis de umbral de color implicó el uso de software para definir umbrales que permitieran distinguir entre corales vivos y blanqueados en las imágenes.

La clasificación visual, en cambio, fue realizada por expertos en coral, quienes examinaron las imágenes para determinar la cobertura y el blanqueamiento coralino.

Los resultados mostraron que cada técnica tenía sus ventajas y limitaciones. El método de cuadrícula de puntos fue altamente preciso en la determinación de la cobertura coralina y el blanqueamiento, pero consumió mucho tiempo para analizar grandes áreas. El análisis de umbral de color fue más

rápido, pero menos preciso, especialmente en áreas con corales oscuros o sombreados. La clasificación visual por expertos fue precisa, pero sujeta a la subjetividad del observador y también requirió mucho tiempo.

Finalmente, como conclusión, se recomendó utilizar una combinación de técnicas para obtener estimaciones más precisas y eficientes de la cobertura coralina y el blanqueamiento en estudios de monitoreo de arrecifes de coral. Esto permitiría aprovechar las fortalezas de cada método y mitigar sus limitaciones, mejorando así la calidad de los datos recopilados para la gestión y conservación de los arrecifes de coral [22].

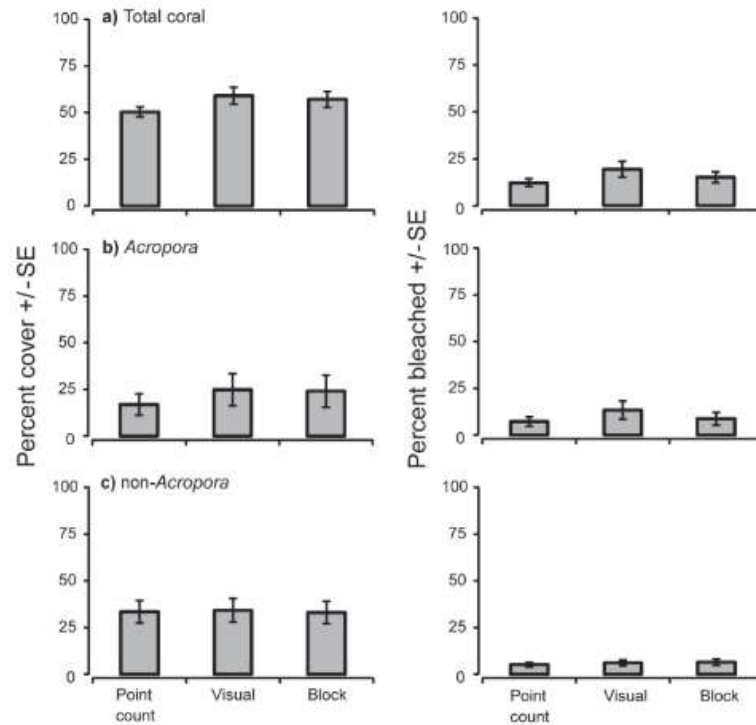


FIGURA 5: Estimación del porcentaje de coral sano y coral blanqueado para los tres métodos estudiados [22]

5. Marco teórico

5.1. Conceptos básicos de redes neuronales artificiales

Las redes neuronales artificiales (RNA) son una representación simplificada del cerebro humano, ya que, al igual que este, está conformado por una red de neuronas interconectadas las cuales transmiten información con el fin de resolver problemas con base a la experiencia, mediante un algoritmo de aprendizaje [30]. La estructura de una red neuronal está compuesta por una capa de entrada, la cual recibe la información del exterior, una capa oculta que intercambia información con las capas internas de la red (puede haber más de una capa oculta) y capa de salida, la cual proporciona la respuesta de la red y transmite la información al exterior [30]. Estas capas, a su vez, tienen consigo un conjunto de elementos y funciones, tal como se describe en el siguiente apartado.

5.1.1. Elementos y funciones de una red neuronal

La unidad de procesamiento que conforma una red neuronal se denomina neurona, la cual es básicamente una función que modela matemáticamente el funcionamiento de la neurona biológica humana de forma aproximada [34], estas neuronas se agrupan por capas que se interconectan entre sí, estableciendo toda una estructura [25], es por ello que para comprender el funcionamiento de toda la red, el modelo de la neurona se describe en la Figura 6

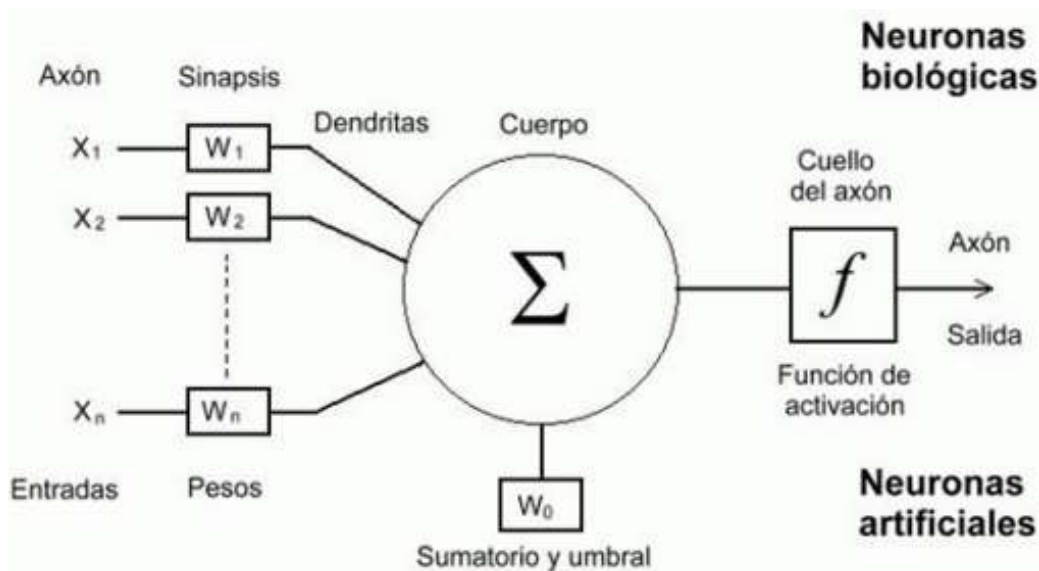


FIGURA 6: Modelo básico de una neurona [34]

El modelo está conformado por un conjunto de entradas X_i las cuales reciben la información numérica de lo que ingresará el sistema, a estas entradas se le asignan los pesos W_i , que representan la sinapsis, estos pesos modifican cada entrada, ya sea reforzándola o debilitándola, dependiendo de la relevancia que tenga cada entrada en el problema y en el proceso de aprendizaje [20], esto se realiza mediante un producto escalar entre el vector de entradas con el de pesos y a este se le añade un umbral W_0 que es el valor a partir del cual se activará la neurona [34]. Finalmente, se hace la sumatoria de cada uno de los productos con el umbral, obteniendo la función de entrada de la neurona:

$$EntradaNeta = \sum_{i=1}^n x_i + w_i + w_o = x_1w_1 + \dots + x_nw_n + w_o \quad (1)$$

La activación de la neurona tendrá lugar una vez el umbral sea sobrepasado y cuando esto suceda, se aplicará una función de activación f a la función de entrada, cuya función es poder concatenar las demás neuronas, esto permite que la red pueda considerar todos los elementos del problema y la relación que hay entre ellos [34], esta expresión será la salida de la neurona:

$$y = f(EntradaNeta) = f(x_1w_1 + \dots + x_nw_n + w_o) \quad (2)$$

Las funciones de activación más comunes en las redes neuronales son:

Función sigmoide La salida de esta función está entre el intervalo de 0 a 1 por lo que es muy usada cuando se necesita predecir una probabilidad [10], modela muy bien la curva de aprendizaje de cualquier red, sin embargo, tiene un costo computacional elevado, por lo que no es recomendable usar esta función para redes muy robustas [34].

$$f(x) = \frac{1}{1 + e^{-x}} \quad (3)$$

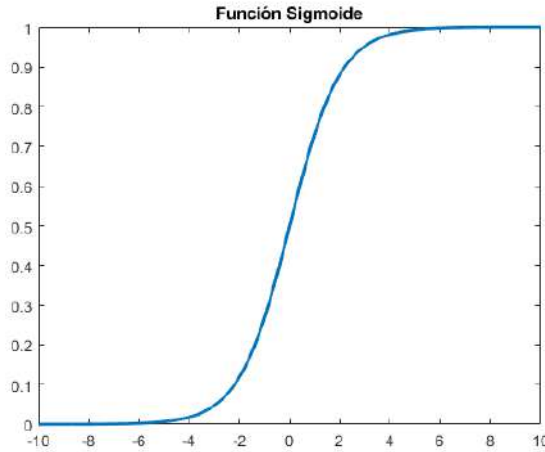


FIGURA 7: Función sigmoide

Función tangente hiperbólica Es una función continua que se comporta de forma similar a la función sigmoide, sólo que sus valores de salida estarán en el rango entre -1 y 1 [10].

$$f(x) = \frac{2}{1 + e^{-2x}} - 1 \quad (4)$$

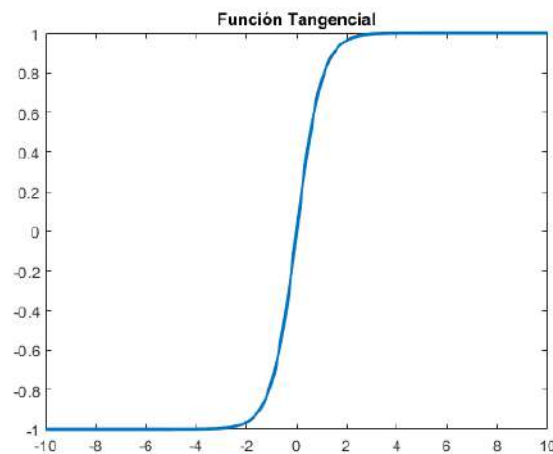


FIGURA 8: Función tangente hiperbólica

Función RELU (Rectified Linear Unit) Es de las funciones más usadas actualmente, especialmente para redes neuronales convolucionales y de aprendizaje profundo [10], ya que tiene un costo computacional bastante bajo en comparación con las demás funciones. Solo obtiene una salida para los valores positivos de la función [34].

$$f(x) = \max(0, x) \quad (5)$$

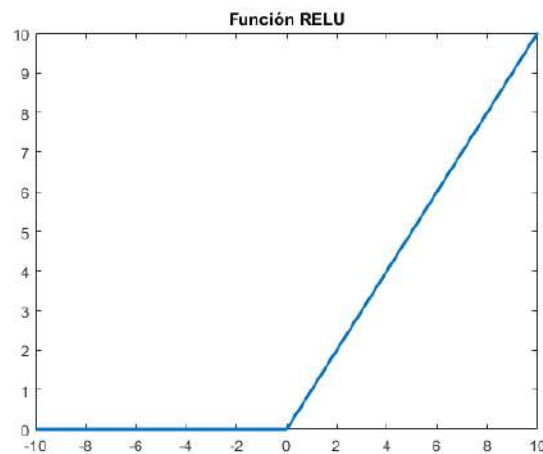


FIGURA 9: Función RELU

5.1.2. Métodos de aprendizaje

El proceso de entrenamiento de una red neuronal, se da cuando esta obtiene una salida a partir de un conjunto de datos de entrada tantas veces, hasta obtener una generalización que de la solución adecuada para cualquier conjunto que ingrese. Este aprendizaje se lleva a cabo modificando los pesos de acuerdo a la información de entrada, ya sea mediante el cambio, creación o destrucción de estos pesos en términos numéricos. A medida que se van ingresando nuevos datos de entrada,

la red va adaptando los pesos hasta un punto en que estos se estabilizarán, esto significa que la red logró obtener una generalización del problema que está tratando la red y por ende, ha aprendido [30].

La manera en que se modifican los valores de los pesos y los criterios usados para ello, depende del tipo de aprendizaje que se use. Los dos métodos principales son el aprendizaje supervisado y el no supervisado.

Aprendizaje supervisado Para este tipo de aprendizaje se requiere de un supervisor que controle y determine la salida que debería obtenerse de acuerdo a información de entrada, el supervisor evalúa si la salida es la deseada o no, si no llega a ser la deseada, se modifican los pesos hasta obtenerla [30].

Aprendizaje no supervisado Este aprendizaje no requiere de un ente externo que evalúe y controle la salida para determinar si se está obteniendo lo deseado, en vez de ello, la red únicamente conoce el espacio de entrada y al abstraer relaciones entre los datos de entrada, se hacen agrupaciones y/o asociaciones para obtener las salidas [30].

5.1.3. Topologías de redes neuronales

Existen diferentes topologías de redes neuronales, dependiendo de la forma en que sus neuronas son interconectadas y de la forma en que procesan la información [38]. Entre las topologías más comunes y utilizadas están la red neuronal feedforward, red neuronal recurrente y red neuronal convolucional.

Red neuronal Feed Forward En este tipo de red, la información fluye de entrada a salida en una única dirección hacia adelante, sin tener alguna clase de bucle o retroceso, su topología es de las más sencillas y no tienen ninguna limitación en cuanto a la cantidad de capas, de neuronas ni los tipos de función de activación que se desean utilizar [38].

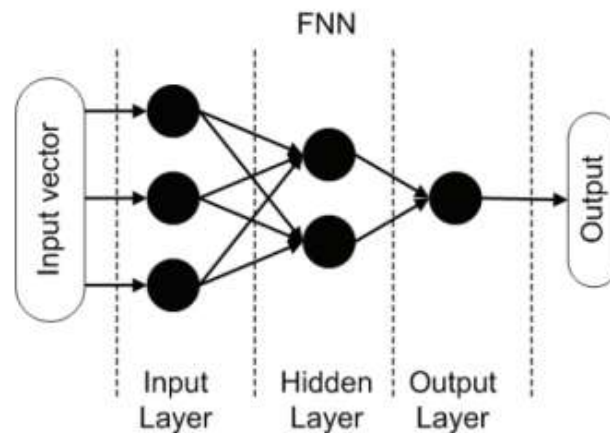


FIGURA 10: Topología red neuronal feedforward [38]

Red neuronal recurrente De manera similar a las redes *feedforward*, no tiene limitaciones en su topología, sin embargo, en este caso la información ya no fluye en un solo sentido, sino que también se transmite hacia atrás, funcionando como una especie de memoria, ya que toma salidas anteriores

como entrada y hace que sus resultados dependan de elementos anteriores, teniendo una especie de comportamiento temporal [38].

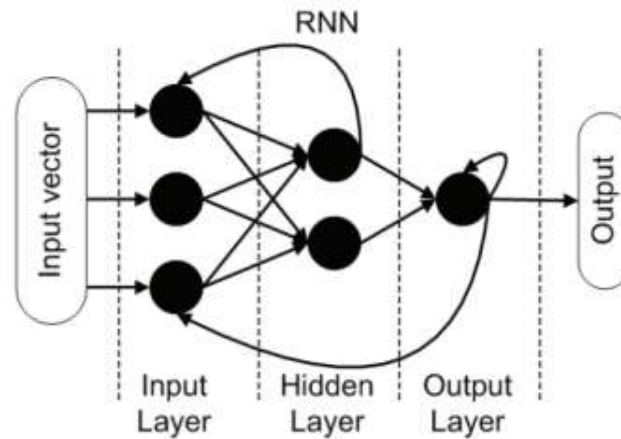


FIGURA 11: Topología red neuronal recurrente [38]

Red neuronal convolucional La especialidad de este tipo de red es analizar imágenes, ya que la entrada se ingresa como una matriz compuesta por píxeles que se procesan para obtener información, para ello se analiza el píxel en cuestión junto con los de su entorno [34], por lo que el procesamiento de información de este tipo de redes es más complejo que los anteriormente mencionados. Básicamente, está compuesto por tres capas, las cuales puede haber más de una para cada tipo, estas son: Convolucional, de agrupación y totalmente conectadas [14].

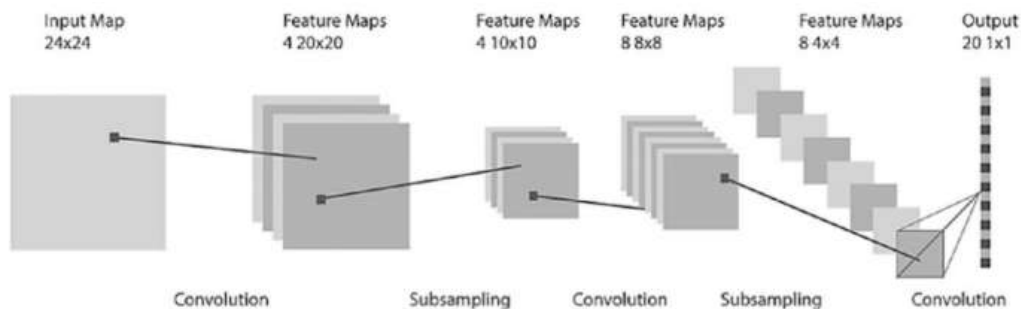


FIGURA 12: Arquitectura de la red neuronal convolucional [11]

La capa convolucional se encarga de capturar las características más generales de la imagen de entrada, esto lo hace tomando vecindades de esta y mediante la convolución calcula un nuevo mapa de características, reduciendo el número de datos a procesar [14].

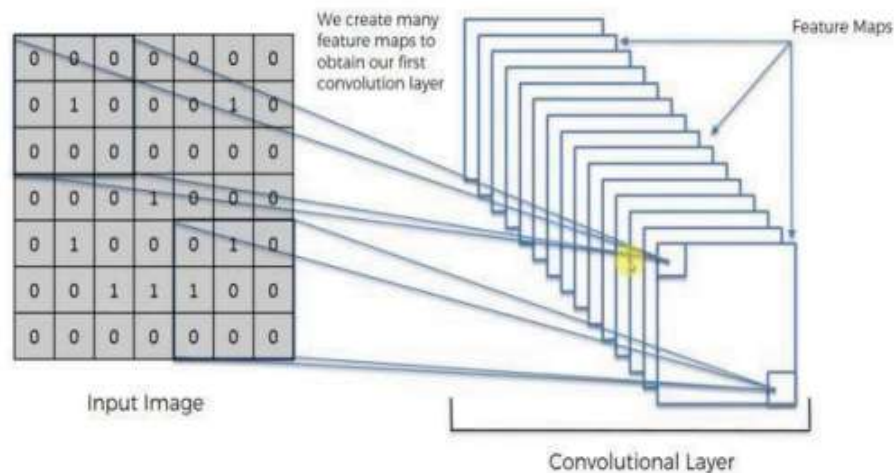


FIGURA 13: Capa convolucional [6]

La capa de agrupación usa un filtro para reducir aún más las dimensiones del mapa de características, dejando las más relevantes para que sean aprendidas por la red [6]. Hay varios tipos de agrupación, entre estos están la agrupación como máxima, la cual es la más usada, la agrupación promedio, agrupación estocástica, entre otras [6].

Finalmente, la capa totalmente conectada, sintetiza toda la información obtenida de las capas anteriores, esta capa está configurada de manera similar a las redes tradicionales, donde las neuronas de la capa actual están conectadas a la capa anterior [7], la última de estas capas es la capa de salida de la cual se obtiene el resultado de la clasificación.

5.2. Aplicaciones de Redes Neuronales Artificiales para el análisis de imágenes

5.2.1. Coral species identification with texture or structure images using a two-level classifier based on Convolutional Neural Networks

Se han propuesto modelos de redes neuronales artificiales para poder clasificar los corales, tal como se estudió y se realizó en “Coral species identification with texture or structure images using a two-level classifier based on Convolutional Neural Networks” [15], en vista de que las diferentes imágenes de corales pueden ser de la estructura general del coral o únicamente de la textura, tal como se muestra en la Figura 14, plantean un clasificador de dos niveles, uno para determinar si la imagen de entrada es una textura o una estructura, y el segundo nivel, dependiendo del tipo de imagen que ingresa, clasificará el tipo de coral. Para ello experimentan con tres arquitecturas de redes neuronales convolucionales, las cuales son Inception, ResNet y DenseNet, haciendo variaciones en su número de capas y de épocas para a partir de ello escoger el mejor modelo con su mejor arquitectura, para así finalmente construir el clasificador. En los resultados se llegó a que la arquitectura ResNet tuvo los mejores resultados para ambos niveles.

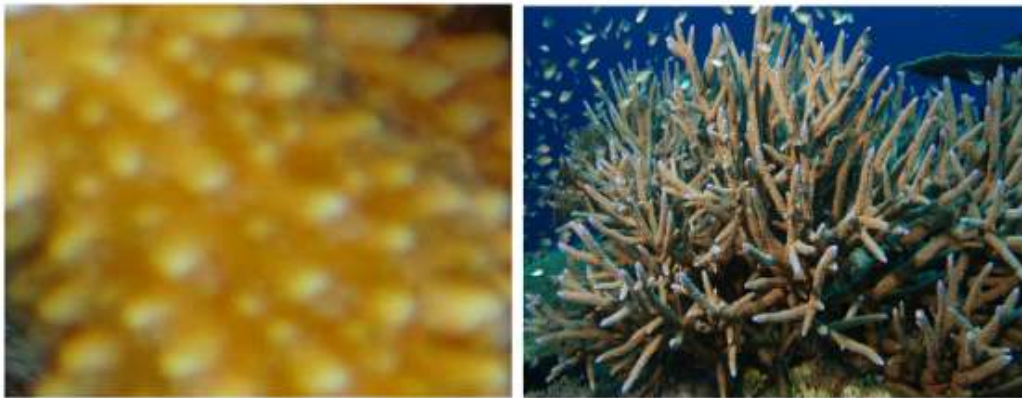


FIGURA 14: Diferencia entre una imagen coralina de textura y estructura. Izquierda, imagen de textura; derecha, imagen de estructura [15]

5.2.2. Recognition of Coronary Arterial Stenosis Using Neural Network on DSA System

En este artículo, se lleva a cabo una red neuronal que permite reconocer la estenosis arterial mediante imágenes de Angiografía por Sustracción Digital.

A partir de esas imágenes realizan un preprocesamiento para obtener las imágenes con el mínimo ruido debido a los latidos del corazón, que no permiten caracterizar adecuadamente los vasos sanguíneos. Una vez teniendo las imágenes con el mínimo error de visualización se extraen las características principales que presentan los vasos, ya que a partir de ellas pueden determinar la estenosis, para ello proponen un algoritmo para el seguimiento de los vasos y así determinar su ubicación, dirección, longitud y diámetro para identificar sus bordes y finalmente obtener las entradas de la red.

En cuanto a la arquitectura de la red neuronal, se implementa una red de 3 capas con número de neuronas por capa de 10-5-1, se usa una función no lineal en la capa oculta y una función lineal en la capa de salida para obtener una respuesta análoga y mejorar la eficiencia de aprendizaje, debido a ello, proponen un nuevo algoritmo Back Propagation (BP) para que pueda estimar este tipo de valores de salida.

Las entradas son los diámetros de los vasos, sus áreas transversales e información de la vecindad, cuyos valores son normalizados. A partir de estas entradas, se determina a la salida el nivel de estenosis, el cual también se normaliza para determinarse de manera porcentual de 0 a 100, una parte de estos datos se destinan para el entrenamiento y otra parte para la evaluación de la red.

Finalmente, luego del entrenamiento y evaluación de la red, se obtuvo un error cuadrático medio de 0.001 con un rango de 100.000 a 190.000 ciclos de aprendizaje [39].

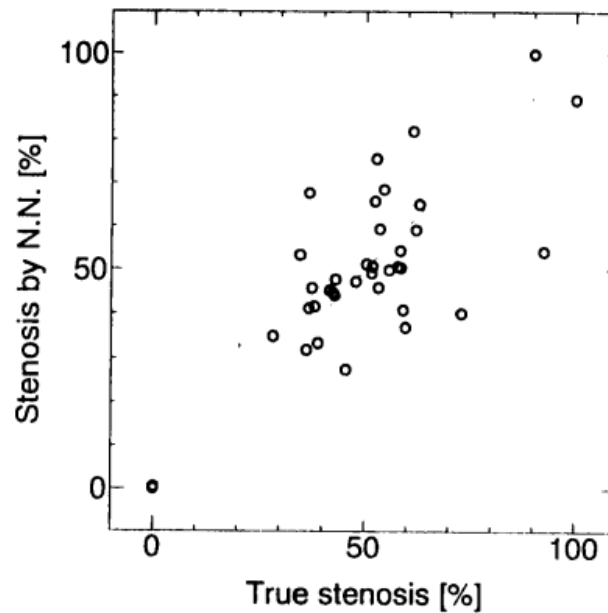


FIGURA 15: Resultados del reconocimiento de estenosis [39]

5.2.3. Three-dimensional medical image analysis of the heart by the revised GMDH-type neural network self-selecting optimum neural network architecture

En este trabajo, se propone un modelo de red neuronal que permita reconocer y extraer los contornos y regiones tridimensionales del corazón a partir de imágenes tomográficas computarizadas de rayos X. Para ello, hacen uso de un algoritmo de red neuronal de tipo GMDH, el cual genera de manera automática, la arquitectura más óptima de red neuronal, al igual que las características de entrada más relevantes para realizar la tarea propuesta, con el fin de minimizar el error de predicción determinado como la predicción de suma de cuadrados PSS.

En vista de que las imágenes diagnósticas médicas presentan características complejas y diferentes entre sí, dependiendo la región a observar y el tipo de imagen que se desea analizar; proponer una arquitectura adecuada puede ser una tarea compleja y propensa al error, lo mismo aplica al determinar qué características tomar como entradas de la red, lo cual no es óptimo para los procesos de diseño. Es por ello que para optimizar dicho proceso y minimizar errores, proponen el algoritmo GMDH el cual utiliza un método de autoorganización heurística que se encargará del diseño de la red.

Se generan múltiples combinaciones de las variables de entrada y de las combinaciones más útiles se obtendrán las arquitecturas más óptimas para la red. Luego de este procedimiento se obtiene que las características de entrada son las coordenadas x y y de los centros de las vecindades, las estadísticas de las densidades de imagen, las cuales son la media, la desviación estándar y la varianza, es decir, cinco variables de entrada. El valor de salida puede ser uno o cero indicando si el píxel en cuestión es una región del corazón o no respectivamente, de acuerdo a ello, el píxel se pinta de color blanco o negro.

En cuanto a la estructura, la red neuronal resultante tiene 4 capas con número de neuronas 5-5-5-1 respectivamente.

Comparando la imagen original que se usó para ser analizada a través de la red neuronal obtenida, con la imagen resultante de ella 16, se puede evidenciar que es un modelo útil para el análisis de este tipo de imágenes médicas [24].

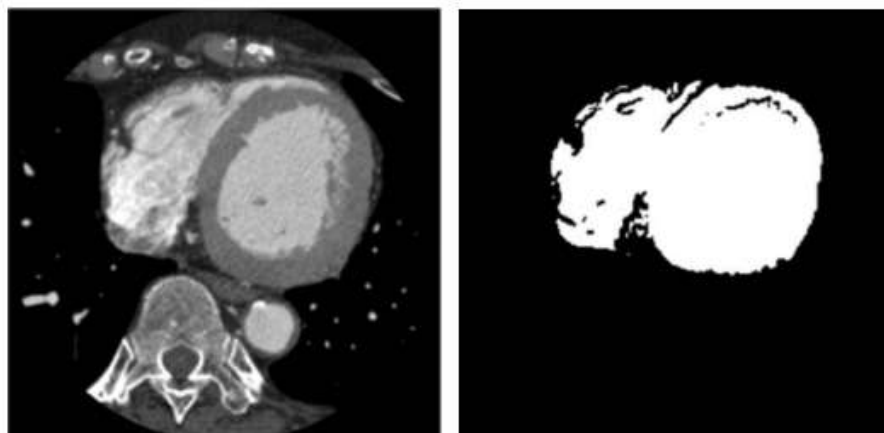


FIGURA 16: Análisis de imágenes tomográficas computarizadas de rayos X del corazón: Izquierda, imagen original; derecha, imagen de salida de la red neuronal [24]

5.3. Técnicas de procesamiento digital de imágenes para la detección de forma y color

5.3.1. Segmentación por color

Esta técnica busca un umbral de separación de cada tonalidad haciendo uso de algoritmos e histogramas, que calculan y separan los componentes de color que posea la imagen de acuerdo al espacio de color que se utilice (RGB, HSL, CIELAB, entre otros).

Una vez se haya calculado un umbral que sirva para etiquetar las tonalidades presentes en la imagen, se puede identificar el color en particular que se desea detectar. Una desventaja que presenta esta técnica, es que el cálculo del umbral que pueda separar los colores depende mucho de la imagen y se puede ver afectado cuando los objetos no son claramente separables, no tienen los colores bien definidos o los rangos de colores son muy similares [33].

En 17 y se puede ilustrar dos ejemplos, donde en el primero se logró umbralizar correctamente el color verde en la imagen; sin embargo, en el segundo ejemplo, a pesar de que se logra umbralizar el color de interés el cual es el morado, se puede ver que también salieron algunos píxeles del color rosado, debido a que sus rangos de color son similares.

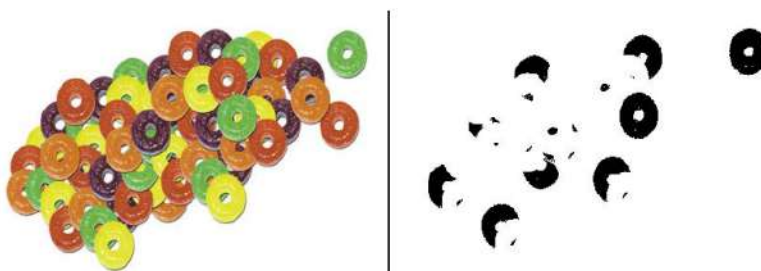


FIGURA 17: Ejemplo de segmentación por color: tuercas de colores [33]



FIGURA 18: Ejemplo de segmentación por color: dulces de colores [33]

5.3.2. Filtros morfológicos

Es un método de procesamiento de imágenes que se basa en la descripción geométrica de los objetos de la imagen. A partir de este principio hay diferentes operaciones de filtrado que realizan diferentes funciones basadas en la forma, las operaciones más básicas en este tipo de filtros son Erosión y Dilatación.

El filtro Erosión, elimina los píxeles vecindarios de un objeto, reduciendo su tamaño o en su defecto, si son objetos muy pequeños, los elimina. Por otra parte, el filtro Dilatación realiza lo opuesto, el cual hace que los píxeles vecindario de un objeto, se vuelvan parte del objeto aumentando su tamaño. A partir de la combinación de estos surgen los filtros de Apertura y Cierre, donde el de Apertura suaviza contornos y el de Cierre rellena agujeros en los objetos [36].

A partir de la combinación y el encadenamiento de estos filtros se pueden realizar algunas aplicaciones que sirvan para la detección de objetos, a continuación se contempla una aplicación y en este caso es la extracción de objetos.

Para extraer un objeto en una imagen primero se define la estructura de dicho objeto en tamaño mayor y menor al original como puntos de referencia, una vez definidas se aplican filtros de apertura en la imagen para eliminar todos los objetos más grandes, teniendo como la estructura de tamaño mayor como punto de referencia y luego, se vuelven a aplicar los mismos filtros para eliminar los objetos más pequeños del objeto en cuestión, esta vez como referencia la estructura de tamaño menor.

Luego de cada filtro habrá una imagen resultante, por lo que la diferencia de estas imágenes dará el resultado, finalmente y en caso de ser necesario se pasa por una umbralización el cual extraerá completamente el objeto [36]. A continuación se ilustra un ejemplo donde se pretenden extraer los autobuses en una imagen de una zona urbana.

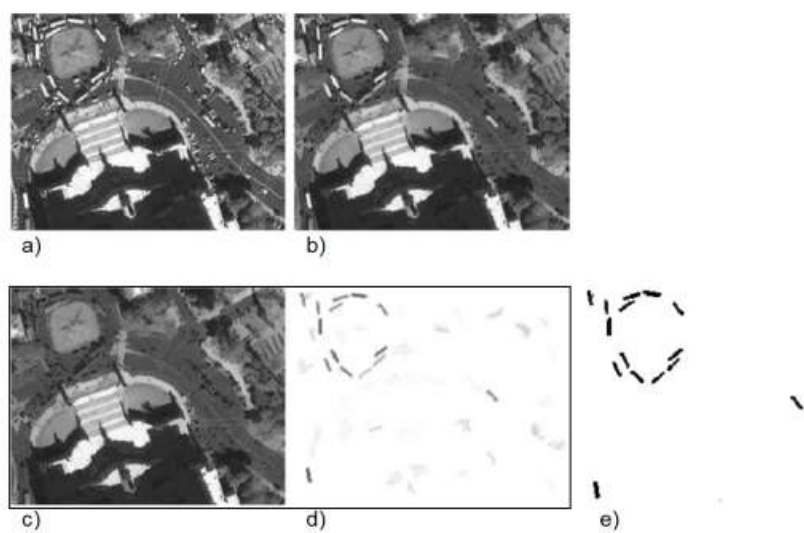


FIGURA 19: Extracción de autobuses en una imagen con filtros morfológicos [36]

6. Metodología

La metodología diseñada para desarrollar cada uno de los objetivos y consistió en cuatro fases.

En la primera fase se exploraron modelos de redes neuronales que realicen la tarea de detección de objetos y segmentación de imágenes, de ellos se seleccionó un modelo de cada uno, teniendo en cuenta varios criterios de selección tales como su desempeño, robustez, capacidad de generalización y su adaptabilidad para ser implementado e integrado a las otras fases del modelo de este proyecto, conformando la red base inicial. Una vez se integraron estos dos modelos a modo de arquitectura en serie donde primero se realice detección del coral y una vez detectado, se segmente del resto de la imagen, se realizó una prueba de la arquitectura tal como está para tener una primera vista de cómo responde la red.

La segunda fase consistió en la parametrización de la arquitectura seleccionada, teniendo en cuenta los resultados de la prueba inicial, para identificar las variables y funciones clave de la red y de esta forma modificarlas en aras de afinar su funcionamiento para la tarea específica con imágenes subacuáticas de corales, ya que la arquitectura inicial tenía una configuración generalizada para el análisis de imágenes y la idea era especializarla hacia lo que se realizó en este proyecto.

En la tercera fase se implementó a la salida y en serie de la red obtenida de la fase dos, un algoritmo que se encargó de la segmentación del color, para que identifique específicamente las regiones blanqueadas y sanas del coral y de esta forma tener un valor numérico porcentual de blanqueamiento, el cual es el dato clave que se quiere extraer de cada imagen.

Finalmente, en la cuarta fase, se realizaron pruebas con la arquitectura resultante y se obtuvieron resultados y valores cuantitativos de error, para observar su desempeño en cuanto a la detección y segmentación de los corales y su respectivo blanqueamiento.

Para el proceso de pruebas y observación de resultados, fue necesario la búsqueda de un banco de imágenes subacuáticas de corales que represente variedades de especies de corales. Además de ello, a cada imagen se le tuvo que realizar un proceso de segmentación manual, donde haciendo uso de un editor de imagen se segmentó lo que observamos como coral, junto con sus regiones blanqueadas y sanas, todo esto bajo la supervisión de un biólogo marino experto en corales, para que esta segmentación se haya realizado de la manera más aproximada a la realidad y tener unas buenas segmentaciones manuales de referencia junto a sus valores porcentuales de blanqueamiento, que se requirieron para comparar con los resultados del modelo y cálculos de error.

Luego de este proceso se analizaron los resultados para identificar y exponer las respectivas conclusiones del proyecto.

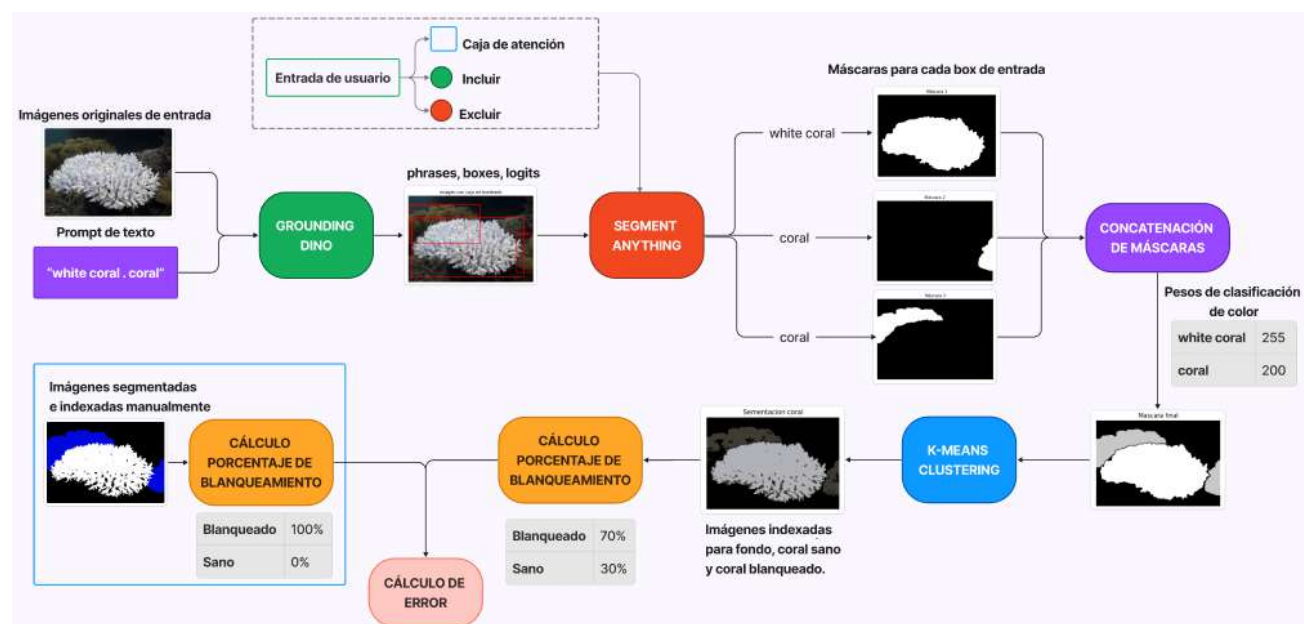


FIGURA 20: Arquitectura propuesta para la solución del problema

7. Desarrollo

7.1. Base de datos seleccionada

La base de datos empleada en este proyecto consiste en un banco de 37 imágenes de corales en total. Este conjunto se compone inicialmente de 12 imágenes capturadas en Colombia, específicamente en Isla Grande, ubicada en el departamento de Cartagena, y en la Isla de Providencia, perteneciente al archipiélago de San Andrés, Providencia y Santa Catalina, durante el año 2023. Las dimensiones de estas imágenes alcanzan un máximo de 350 píxeles de ancho por 350 de alto, en formato JPG, y fueron tomadas con el propósito de un posible estudio del blanqueamiento de los corales.

Adicionalmente a las imágenes mencionadas, se dispone de otro conjunto de 25 imágenes obtenidas de una base de datos de uso libre, específicamente seleccionadas para la clasificación de corales sanos y blanqueados. Esta base de datos puede ser accedida en Kaggle a través del siguiente enlace [3]. Dichas imágenes se clasifican en dos categorías distintas: corales sanos (438 imágenes) y corales blanqueados (485 imágenes). El formato de estas imágenes es JPG y su tamaño máximo es de 350 píxeles de ancho y alto, con el fin de mantener una consistencia en el tamaño a lo largo del conjunto de datos. No obstante, para este proyecto, la variabilidad en los tamaños no resulta relevante para el rendimiento del modelo.

La razón detrás de la selección de muestras de esta última base de datos radica en la amplia variedad de especies de corales que ofrece, debido a la gran cantidad de imágenes que la componen. Esto proporciona una diversidad de especies de coral mayor que la obtenida inicialmente, lo cual es fundamental para evaluar el desempeño del modelo propuesto en este proyecto.

Es importante destacar que en este caso no fue necesario contar con una base de datos excesivamente extensa con el fin de obtener un mejor aprendizaje. Esto se debe a que, al tratarse de modelos pre-entrenados como Grounding DINO y Segment Anything, entrenados con grandes y diversos conjuntos de imágenes, el desafío principal no reside en el aprendizaje de una única imagen, sino en la parametrización y configuración para que ejecuten las tareas específicas requeridas en cada caso. Por otro lado, tampoco se recopiló un gran número de imágenes debido a que se requiere segmentación manual, lo cual es una tarea bastante ardua, con la asistencia de un experto para cada una de ellas. En consecuencia, el límite de 35 imágenes fue el máximo con el que el experto nos proporcionó su ayuda.

7.1.1. Proceso de segmentación manual

La segmentación manual de las imágenes se llevó a cabo mediante el uso del programa de edición de imágenes Gimp, con el propósito de realizar una detallada selección de las distintas áreas de interés: el coral sano, la región blanqueada del coral y el fondo circundante. Este proceso se llevó a cabo utilizando las herramientas de selección proporcionadas por el programa. Una vez definidas estas tres regiones, se asignaron colores específicos a cada una:

- Se utilizó el color azul para representar el coral sano.
- Se empleó el color blanco para identificar las regiones blanqueadas del coral.
- El color negro se asignó al fondo de la imagen.

Cada imagen de la base de datos fue sometida a este proceso, y posteriormente se presentaron al experto para su evaluación. El experto determinó la calidad de la segmentación realizada, señalando

cualquier área que requiriera corrección. A partir de estas indicaciones, se efectuaron las correcciones pertinentes hasta obtener imágenes segmentadas que cumplieran con los criterios de referencia establecidos.

Además, utilizando el mismo programa de edición de imágenes, todas las imágenes segmentadas fueron indexadas a tres colores. Esta funcionalidad, disponible en el formato de imagen PNG, permite codificar cada píxel como un único número entero, junto con una tabla de indexación de colores que asigna a cada número entero el color correspondiente. Esta técnica no solo beneficia la compresión del archivo, especialmente en escenarios web, sino que también facilita la manipulación de los datos en Python. Al representar cada píxel como un valor entero, se simplifica la realización de operaciones como la normalización de colores, la comparación de colores y el conteo de píxeles. Estas operaciones son fundamentales para calcular las áreas de coral blanqueado y sano, lo que a su vez permite determinar los porcentajes de blanqueamiento de cada muestra de coral.

7.2. Exploración y selección de modelos de detección y segmentación de imágenes digitales con redes neuronales

7.2.1. Detección de imágenes digitales con redes neuronales

EfficientDet EfficientDet es un modelo de detección de objetos que se destaca por su eficiencia en términos de tamaño del modelo y recursos computacionales, manteniendo al mismo tiempo un alto rendimiento. Introduce varias innovaciones clave, como BiFPN, una red de pirámide de características bidireccionales que permite la fusión rápida y sencilla de características multiescala para una detección precisa de objetos de diferentes tamaños en una imagen.

Además, utiliza un método de escalamiento compuesto que ajusta uniformemente la resolución, la profundidad y el ancho de la red neuronal, manteniendo un equilibrio entre la complejidad del modelo y su rendimiento. Al emplear el backbone de EfficientNet, el modelo se beneficia de una arquitectura eficiente previamente optimizada para la precisión, lo que lo hace adecuado para despliegues en entornos con recursos limitados o aplicaciones en tiempo real donde la eficiencia es crucial. [40]

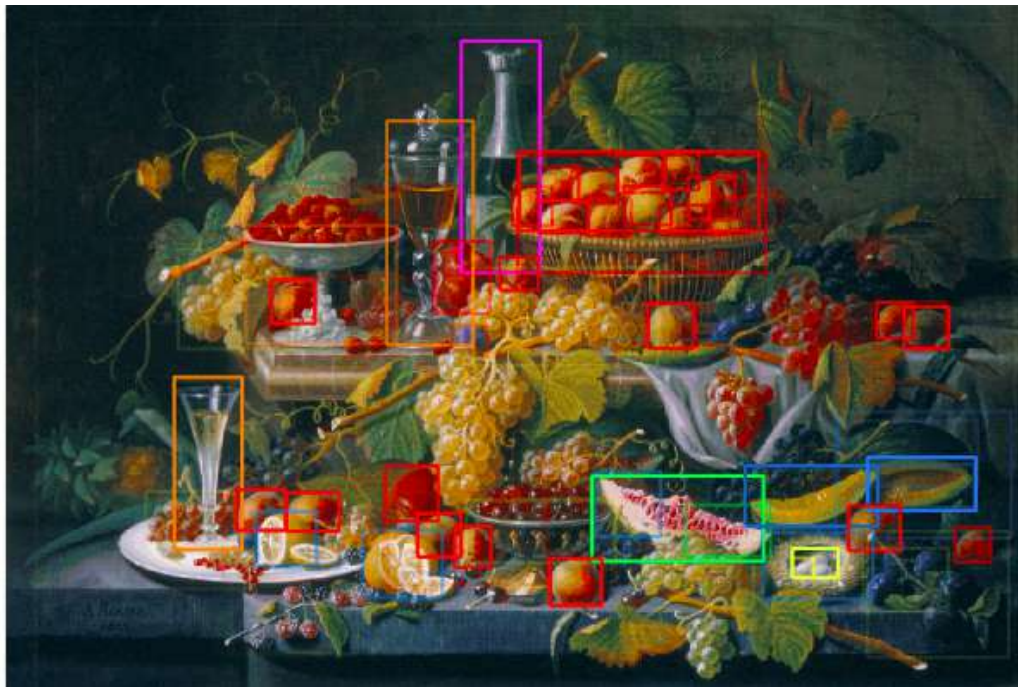


FIGURA 22: OWL-ViT: Detección de objetos en mundo abierto con Vision Transformers [31]

Grounding DINO GroundingDINO es un modelo innovador de detección de objetos de conjunto abierto que revoluciona el panorama de la detección al incorporar entradas de lenguaje para mejorar el reconocimiento de objetos. Alcanzando un rendimiento notable en el conjunto de datos COCO, demuestra una precisión promedio de cero disparos de 52.5 sin utilizar ningún dato de entrenamiento de COCO, con un aumento subsiguiente a 63.0 después de la afinación. Además, GroundingDINO muestra su versatilidad a través de colaboraciones con Stable Diffusion para la edición de imágenes y Semantic-SAM para la segmentación universal de imágenes, capacitando a los usuarios para realizar ediciones y segmentaciones flexibles y precisas.

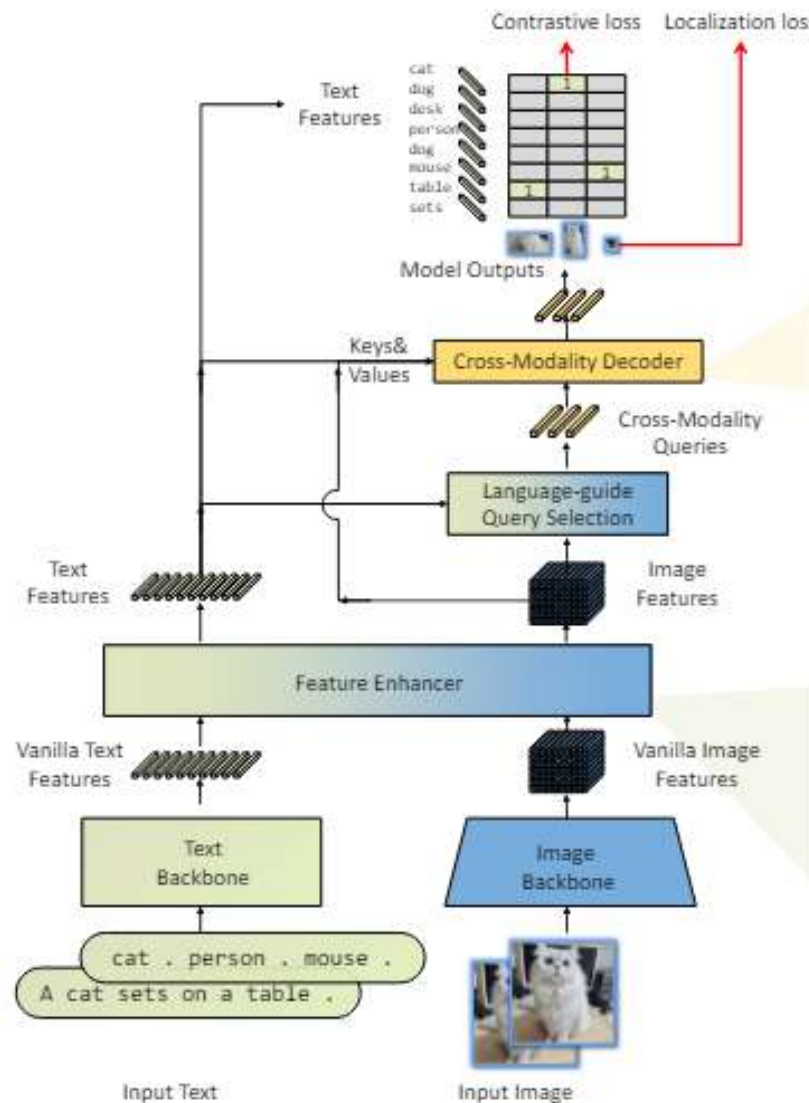


FIGURA 23: Modelo Grounding DINO [27]

Grounding DINO es una arquitectura de codificador dual y decodificador único. Contiene un backbone de imagen para la extracción de características de imagen, un backbone de texto para la extracción de características de texto, un potenciador de características para la fusión de características de imagen y texto, un módulo de selección de consultas guiado por idioma para la inicialización de consultas y un decodificador multimodal para el refinamiento de la caja [27].

Además de su impresionante rendimiento en la detección de objetos, Grounding DINO destaca por su compromiso con la accesibilidad y la colaboración comunitaria. La liberación de su código y demos permite a investigadores y desarrolladores explorar y aplicar esta tecnología en una amplia gama de proyectos y aplicaciones. Su capacidad para funcionar incluso en modo solo CPU amplía su accesibilidad, lo que significa que puede ser utilizado en una variedad de configuraciones de hardware. Esta disponibilidad y enfoque colaborativo fomentan la innovación y el progreso en la comunidad de visión por computadora, al tiempo que promueven un mayor entendimiento y uso de la detección de objetos con Grounding DINO. [27]

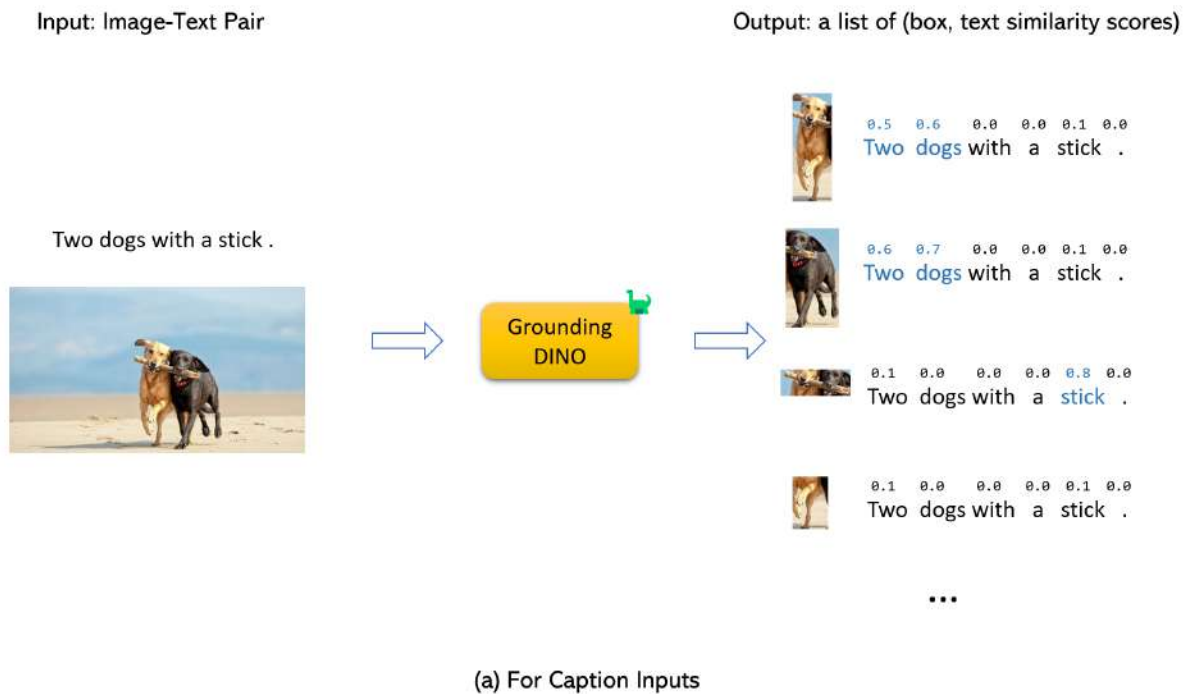


FIGURA 24: Ground DINO: Explicación de las entradas y salidas [27]

MobileNet MobileNet es una familia de modelos de redes neuronales diseñados específicamente para aplicaciones de visión en dispositivos móviles y embebidos, como teléfonos inteligentes, cámaras de seguridad y drones. Estos modelos están optimizados para ser livianos y eficientes en términos de tamaño y velocidad de procesamiento, lo que los hace ideales para entornos con recursos computacionales limitados. La arquitectura de MobileNet se basa en convoluciones separables en profundidad, una técnica que divide las operaciones de convolución en dos etapas: primero, una convolución en profundidad, la cual aplica un filtro a cada canal de entrada, y luego, una convolución punto a punto que combina las salidas de la convolución en profundidad. Esta estrategia reduce significativamente el número de parámetros y la cantidad de operaciones necesarias, manteniendo al mismo tiempo la capacidad de representación de la red.

Además, MobileNet introduce dos hiperparámetros clave: el multiplicador de ancho y el multiplicador de resolución. El multiplicador de ancho controla el número de canales en cada capa de la red, lo que permite ajustar la capacidad del modelo según las necesidades del problema. Un valor más pequeño de este parámetro reduce el tamaño y el costo computacional del modelo, mientras que un valor más grande puede mejorar la capacidad de representación, aunque a expensas de un mayor costo computacional.

Por otro lado, el multiplicador de resolución ajusta las dimensiones espaciales de la imagen de entrada, lo que permite adaptar el modelo para trabajar con imágenes de diferentes resoluciones. Esto puede ser útil para reducir la carga computacional al procesar imágenes de baja resolución o para capturar detalles espaciales finos en imágenes de alta resolución. En conjunto, estos componentes hacen que MobileNet sea una opción versátil y eficiente para una amplia gama de aplicaciones de visión en dispositivos móviles y embebidos. [18]



FIGURA 25: MobileNet: Uso en entornos urbanos [18]

YOLO YOLO (You Only Look Once) quizá es el modelo de detección de objetos más reconocido en el campo. Es un enfoque revolucionario en la detección de objetos que aborda el problema como una tarea de regresión, permitiendo la predicción directa de cajas delimitadoras y probabilidades de clase a partir de imágenes completas en una única evaluación de red neuronal. Su arquitectura unificada optimiza el rendimiento de detección de extremo a extremo dentro de una sola red, lo que agiliza significativamente el proceso. Destacando su velocidad, el modelo base de YOLO procesa imágenes en tiempo real a 45 cuadros por segundo, con una versión más pequeña, Fast YOLO, alcanzando asombrosamente 155 cuadros por segundo, mientras mantiene el doble del mAP de otros detectores en tiempo real.

Además de su velocidad, YOLO demuestra una notable capacidad de generalización, superando ampliamente a otros métodos de detección al generalizar desde imágenes naturales hasta obras de arte en diversos conjuntos de datos. YOLOv3, la versión más reciente, destaca por su velocidad y precisión, logrando un mAP comparable con Focal Loss pero siendo aproximadamente cuatro veces más rápido. Esta versión también ofrece la flexibilidad de ajustar el tamaño del modelo para equilibrar la velocidad y la precisión sin necesidad de volver a entrenar el modelo.

YOLO representa un avance significativo en la detección de objetos al proporcionar una combinación única de velocidad, precisión y capacidad de generalización. Su enfoque innovador, que emplea una sola evaluación de red neuronal en la imagen completa, ha revolucionado la detección de objetos y ha establecido un nuevo estándar en el campo de la visión por computadora.[35]

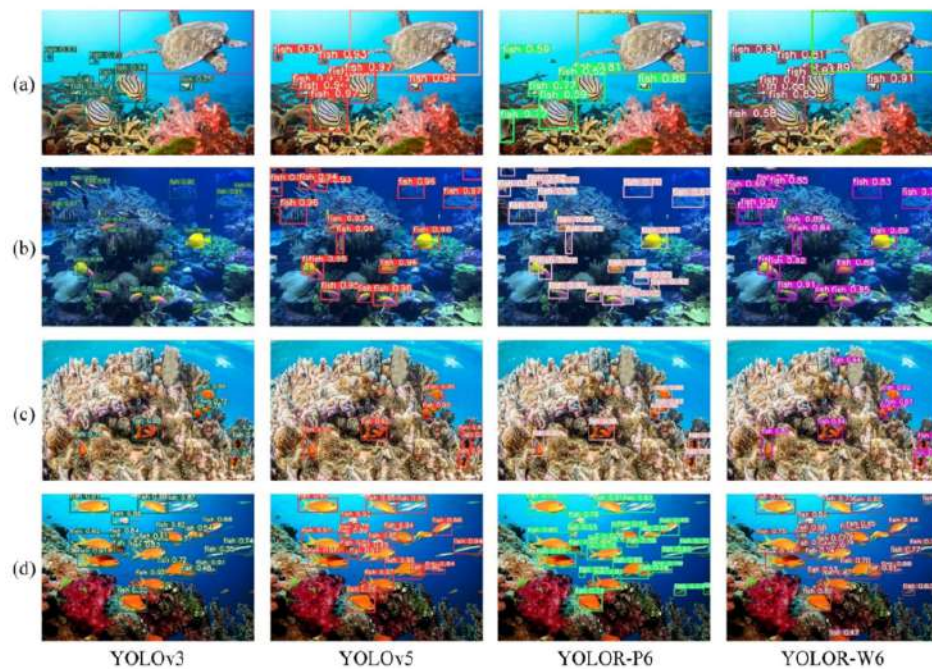


FIGURA 26: Detección de animales marinos en tiempo real utilizando redes de aprendizaje profundo basadas en YOLO en el ecosistema de arrecife de coral [43]

7.2.2. Segmentación de imágenes digitales con redes neuronales

Segment Anything Segment Anything es un innovador proyecto en el campo de la visión por computadora que tiene como objetivo democratizar la tarea de segmentación de imágenes. Introduciendo el Modelo Segment Anything (SAM) y el conjunto de datos Segment Anything 1-Billion (SA-1B), el proyecto busca simplificar el proceso de creación de modelos precisos de segmentación, tradicionalmente reservados para expertos técnicos con acceso a infraestructura de entrenamiento de IA y grandes volúmenes de datos anotados. SAM es un modelo eficiente entrenado en un amplio conjunto de datos de 11 millones de imágenes y 1,1 mil millones de máscaras, capaz de generar máscaras de objetos de alta calidad a partir de diversos tipos de entrada, como puntos o cuadros.

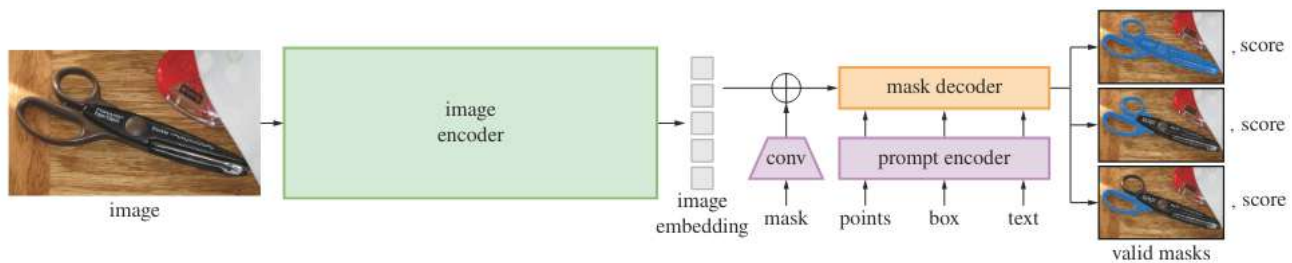


FIGURA 27: Modelo Segment Anything (SAM) [23]

SAM tiene tres componentes, ilustrados en 27 un codificador de imágenes, un codificador de prompts flexible y un decodificador de máscara rápida. El codificador de imágenes genera una imagen incrustada que luego puede consultarse de manera eficiente mediante una variedad de mensajes de

entrada para producir máscaras de objetos a una velocidad amortizada en tiempo real. Para mensajes ambiguos correspondientes a más de un objeto, SAM puede generar múltiples máscaras válidas y puntuaciones de confianza asociadas [23].

El conjunto de datos SA-1B, el más grande de su tipo hasta la fecha, contiene más de mil millones de máscaras en 11 millones de imágenes, todas con licencia y respetando la privacidad. Este conjunto de datos proporciona una base sólida y diversa para entrenar modelos de segmentación y realizar investigaciones en visión por computadora. A través del enfoque de democratización, Segment Anything busca reducir la necesidad de experiencia especializada en modelado de tareas específicas, recursos de cómputo de entrenamiento y anotación de datos personalizados para la segmentación de imágenes, abriendo así nuevas oportunidades en el campo.

SAM, con su capacidad para adaptarse a nuevas distribuciones de imágenes y tareas, tiene el potencial de ser utilizado en una amplia gama de aplicaciones, desde la edición de fotos hasta la comprensión multimodal del mundo. Además, el proyecto Segment Anything abre la puerta a posibilidades emocionantes en campos como la realidad aumentada, realidad virtual, la edición de contenido creativo y la investigación científica [23], [SAM].

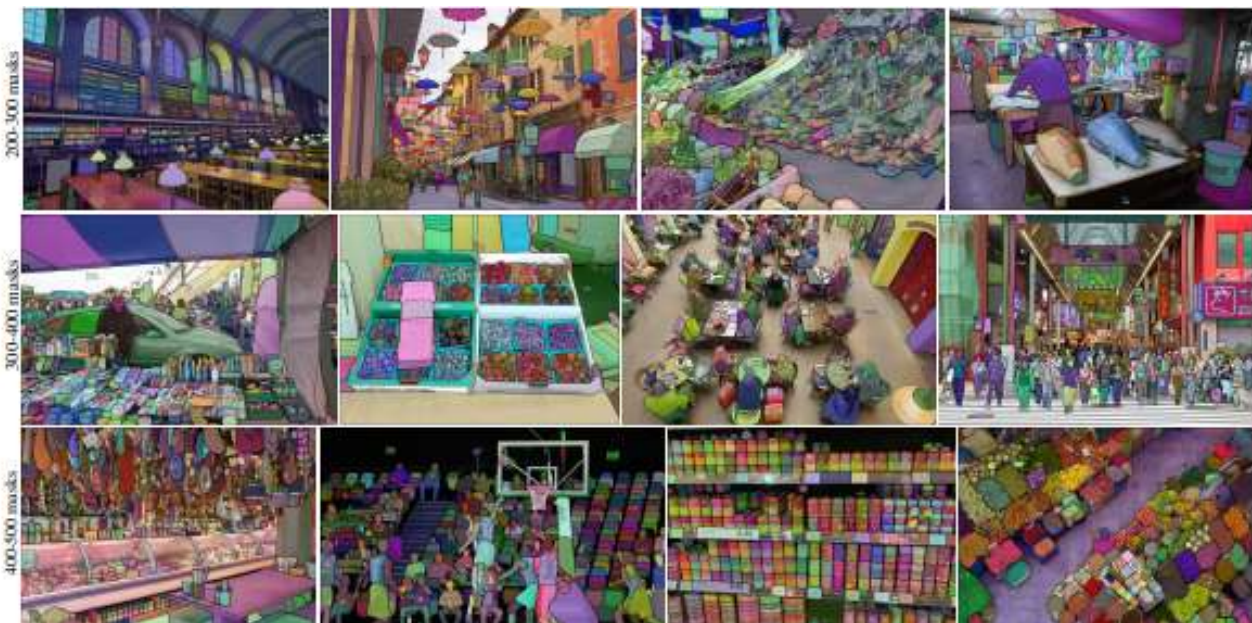


FIGURA 28: Ejemplo de segmentación automática usando SAM, con imágenes del conjunto de datos SA-1B [23]

Segment Everything Everywhere All at Once Es un modelo interactivo y eficiente para segmentar todos los objetos en una imagen simultáneamente. SEEM se basa en una arquitectura codificador-decodificador de transformador simple con un codificador de texto adicional. En lugar de utilizar un enfoque convencional de segmentación que requiere diferentes modelos para diferentes tipos de consultas, SEEM propone un enfoque unificado que puede manejar múltiples tipos de consultas, como puntos, cuadros, garabatos y máscaras, así como consultas de texto.

El proceso de decodificación en SEEM simula un modelo generativo de lenguaje grande (LLM) pero con una interfaz multimodal de entrada y salida. Se utilizan un codificador de imagen y un codificador de texto para convertir todas las consultas en indicaciones visuales y de texto respectivamente, que luego se alimentan al decodificador. Esto permite que el modelo maneje de manera eficiente una variedad de tipos de consultas y promueva la sinergia entre los mensajes visuales y de texto a través de la atención cruzada.

Una característica destacada de SEEM es su capacidad de generalización a consultas novedosas o combinaciones de consultas, lo que lo convierte en una interfaz de segmentación de imágenes universal. Además, el modelo es eficiente para la segmentación interactiva, ya que solo requiere una ejecución del extractor de características al principio y una decodificación ligera en cada ronda de interacción con los humanos [44].

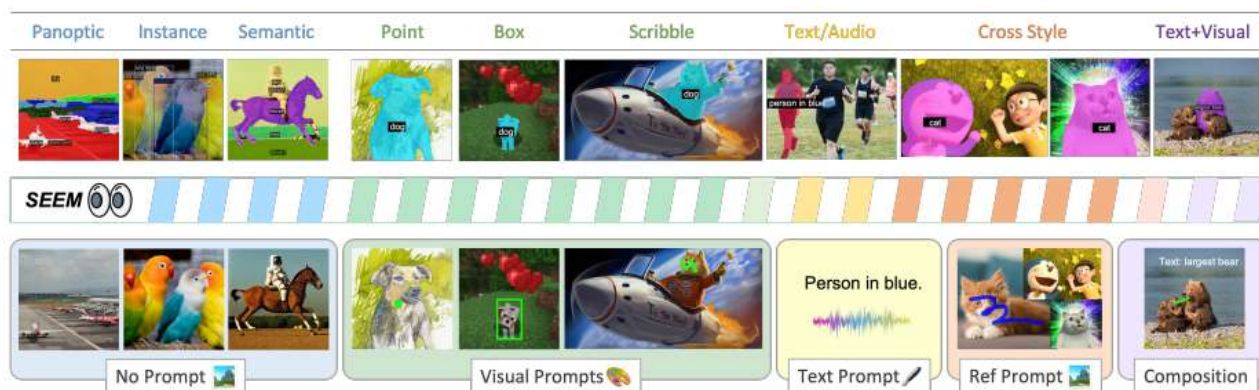


FIGURA 29: SEEM admite tareas de segmentación genéricas de forma abierta cuando no se proporciona ningún mensaje. También permite el uso de indicaciones visuales, textuales y de regiones de referencia.[44]

SegGPT: Segmenting Everything In Context SegGPT es un modelo diseñado para segmentar elementos dentro de imágenes o videos de manera contextual. Unifica diferentes tareas de segmentación en un solo marco de trabajo, permitiendo manejar diversos tipos de datos de segmentación de manera efectiva. Se entrena en un problema de coloreado contextual, utilizando una estrategia de mapeo de colores aleatorios y un objetivo basado en el contexto. Esto le permite adaptarse a diversas tareas de segmentación sin depender de características específicas de la tarea o esquemas de color.

Es capaz de realizar diversas tareas de segmentación, como segmentación de instancias de objetos, segmentación de elementos, segmentación de partes, detección de contornos y segmentación de texto, utilizando inferencia contextual, además, incorpora una estrategia de conjunto de contexto llamada conjunto de características, que combina de manera efectiva múltiples ejemplos en contexto durante la inferencia para mejorar el rendimiento del modelo. SegGPT también puede funcionar como un modelo especializado para tareas específicas, como la segmentación semántica ADE20K, sin necesidad de actualizar sus parámetros, representando un avance significativo en el campo de la segmentación, ofreciendo una solución unificada, flexible y potente para segmentar elementos diversos dentro de datos visuales [41].

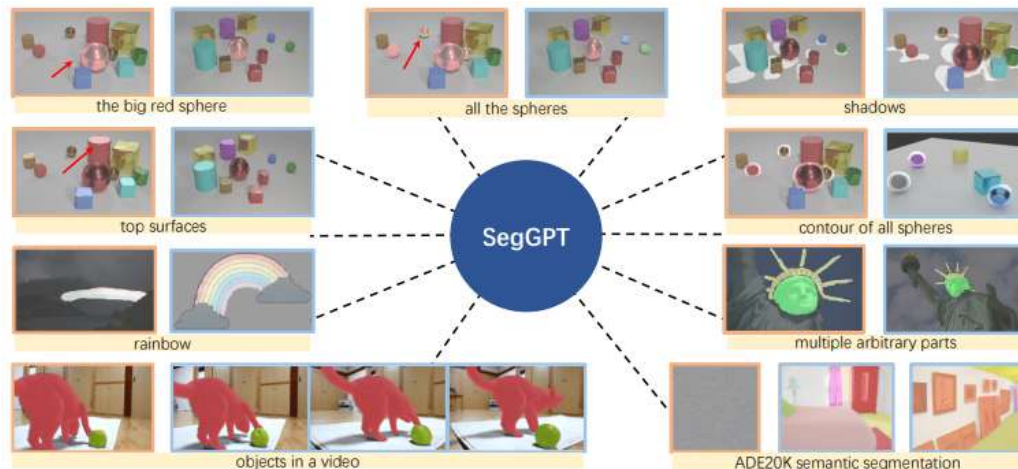


FIGURA 30: SegGPT es capaz de segmentar todo en contexto con un solo modelo, que utiliza ejemplos en contexto para indicar diferentes tareas. Para cada muestra, el cuadro naranja a la izquierda muestra la imagen de ejemplo/solicitud y su máscara correspondiente, mientras que el cuadro azul a la derecha muestra la imagen de entrada y la salida de máscara resultante. La máscara representa la región brillante adjunta a la imagen. El título de cada muestra (en el cuadro amarillo) es solo para explicación.[41]

Deep Multi-instance Convolutional Neural Network (DMCNN) La Red Neuronal Convolutiva de Múltiples Instancias Profundas (DMCNN) es un modelo diseñado para clasificar desastres naturales en imágenes de teledetección de alta resolución. Su enfoque se centra en combinar la extracción de características con el aprendizaje de prototipos para lograr una clasificación precisa. Para ello, utiliza técnicas avanzadas como la Agrupación de Pirámides Espaciales Atroces (ASPP), que amplía el campo receptivo de la capa convolutiva, mejorando así la detección de características en las imágenes. Además, DMCNN emplea una estructura de red neuronal convolutiva (CNN) para extraer y procesar características de las imágenes, lo que le permite identificar características relevantes de los desastres naturales.

En cuanto al aprendizaje de prototipos, DMCNN introduce una capa especializada que asigna características de instancia a prototipos a nivel de bolsa. Esta capa utiliza una medida de distancia para agrupar las características de manera efectiva, mejorando así la capacidad del modelo para clasificar los desastres. La integración de características se lleva a cabo combinando todas las características de instancia, tanto del prototipo como de la bolsa, lo que permite una detección y clasificación exhaustiva de los desastres en las imágenes de teledetección.

Los resultados experimentales muestran que DMCNN supera a los modelos tradicionales de CNN en la precisión de clasificación de desastres naturales. Además, se observa una mejora significativa en el rendimiento de clasificación cuando se utilizan imágenes de detección remota de alta resolución débilmente supervisadas [42].

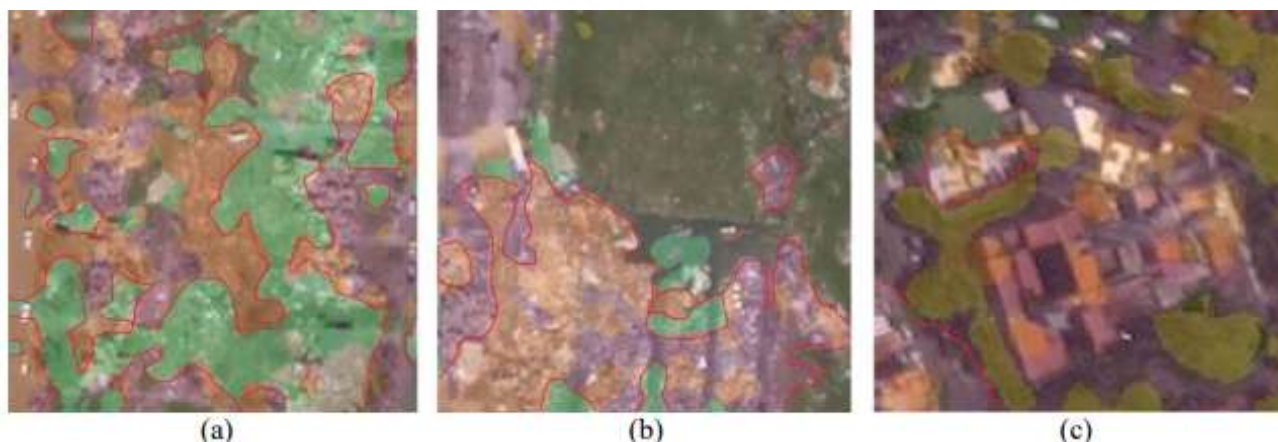


FIGURA 31: Clasificación del desastre del terremoto con diferentes fondos circundantes de objetos terrestres, (a) área de color uniforme, (b) distribución de color compleja y área de contraste grande, (c) distribución compleja de objetos terrestres circundantes. El marco de la línea roja es el resultado dibujado a mano mediante interpretación visual, y las áreas violeta y amarilla son los resultados de clasificación de los desastres sísmicos obtenidos por el modelo DMCNN.[42]

7.2.3. Selección de los modelos

La selección de modelos de IA para abordar la detección de blanqueamiento de corales se basó en varios criterios fundamentales que incluyen precisión, capacidad de generalización, eficiencia computacional y capacidad para manejar diferentes condiciones de iluminación y entornos marinos. Los modelos seleccionados, GroundingDINO y SAM (Segment Anything), cumplen con estos criterios de manera sobresaliente, lo que los hace idóneos para esta tarea específica.

GroundingDINO para reconocimiento de corales:

- **Precisión:** GroundingDINO ha demostrado ser altamente preciso en tareas de reconocimiento de objetos, incluso en entornos marinos complejos. Utiliza una combinación de redes neuronales profundas y aprendizaje por representación para identificar con precisión los corales y distinguirlos de otros elementos del entorno submarino.
- **Capacidad de generalización:** Este modelo ha sido entrenado con un amplio conjunto de datos que incluyen diversas formas y condiciones de iluminación, lo que le permite generalizar eficazmente en diferentes entornos marinos y escenarios de detección.
- **Eficiencia computacional:** A pesar de su complejidad, GroundingDINO es relativamente eficiente en términos de recursos computacionales requeridos, lo que lo hace práctico para implementaciones en tiempo real o en dispositivos con recursos limitados.
- **Adaptabilidad:** GroundingDINO puede adaptarse fácilmente a nuevas especies de corales o condiciones ambientales mediante el ajuste fino o la ampliación del conjunto de datos de entrenamiento.
- **Funcionamiento en modo solo CPU:** La capacidad de GroundingDINO para funcionar eficientemente, incluso en modo solo CPU amplía aún más su accesibilidad, permitiendo su implementación fácil para las diferentes pruebas y creación de la solución.

SAM (Segment Anything) para segmentación de corales:

- **Precisión y robustez:** SAM es conocido por su capacidad para segmentar objetos en imágenes con precisión y robustez, incluso en presencia de variaciones en la iluminación y la textura del fondo marino. El uso masivo de diferentes imágenes en diferentes contextos para su entrenamiento, vuelve a SAM una excelente opción para segmentar imágenes de corales que a priori posiblemente no se hayan entrenado. Esto es crucial para identificar con precisión áreas específicas de coral, incluidas las zonas afectadas por el blanqueamiento.
- **Flexibilidad:** SAM es altamente flexible y puede adaptarse a una amplia gama de tareas de segmentación, lo que lo hace ideal para identificar y delimitar regiones de interés, como los corales en cuestión, descartando elementos que pueden parecer corales, pero son realmente parte del fondo como piedras, animales, acantilados e incluso otros elementos blancos o de color similar a los colores normales de los corales.
- **Eficiencia y velocidad:** Aunque la segmentación de objetos puede ser computacionalmente intensiva, SAM ha sido optimizado para lograr una ejecución eficiente y rápida, lo que lo hace adecuado para aplicaciones en tiempo real o análisis de grandes volúmenes de datos.
- **Capacidad de integración:** SAM es compatible con diferentes arquitecturas de red neuronal y marcos de trabajo de aprendizaje profundo, lo que facilita su integración en sistemas existentes o la combinación con otros modelos de IA, como el ya escogido GroundingDINO.

En conjunto, la combinación escogida de GroundingDINO para reconocimiento de corales y SAM para segmentación de corales proporciona una solución integral y robusta para la detección de blanqueamiento de corales. Estos modelos han sido seleccionados cuidadosamente consultando a varios expertos en el área (profesores en la facultad, compañeros en el mercado de la ciencia de datos, foros en Internet) debido a su capacidad demostrada para abordar los desafíos específicos asociados con esta tarea crítica de conservación marina.

7.2.4. Primeras pruebas

La evaluación inicial del desempeño de SAM (Segment Anything) reveló su eficacia en la segmentación de objetos, destacando su capacidad para abordar una amplia gama de contextos sin una especificidad predefinida. Aunque no privilegia ni asigna ponderaciones a elementos particulares dentro de una imagen, su destreza para segmentar objetos en general es evidente. Este análisis preliminar estuvo restringido al ámbito marino, subrayando la versatilidad de SAM en la segmentación de objetos sin consideraciones contextuales específicas.

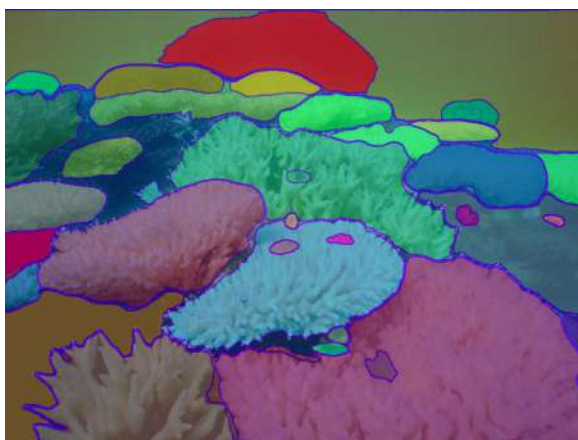
Sin embargo, es relevante destacar que SAM carece de la capacidad intrínseca para atribuir prioridades, etiquetas o pesos a los elementos identificados en una imagen dada. Esta característica implica que, si bien SAM puede segmentar efectivamente los objetos presentes en una imagen, no contextualiza ni distingue la importancia relativa de estos elementos dentro de su entorno específico. En consecuencia, la segmentación realizada por SAM puede considerarse como una acción fundamentalmente independiente de cualquier contexto subyacente, lo que limita su utilidad en la detección específica de corales u otros elementos particulares en un conjunto de imágenes.

La Figura 49 proporciona una representación visual de la capacidad de SAM para segmentar objetos diversos, incluidos los corales, a partir de una imagen de referencia del fondo marina. Se observa

cómo SAM logra segmentar incluso elementos desafiantes, como los corales en primer plano con una frontera difusa y aquellas piedras que se presentan de manera poco clara en el fondo.



(A) Imagen Original



(B) Imagen segmentada

FIGURA 32: Ejemplo segmentación completa sin semántica y con parámetros por defecto

Ante la limitación inicial de SAM para proporcionar contexto en la segmentación de objetos, se decidió investigar alternativas que incorporaran cierto grado de semántica en la detección de objetos, sin necesidad de segmentar la imagen píxel por píxel. Esta decisión se tomó con el objetivo de capitalizar la capacidad de SAM para prestar atención a áreas específicas de una imagen mediante cajas delimitadoras y/o puntos de referencia, permitiéndole discernir con precisión qué objetos debían ser segmentados.

La estrategia adoptada implicó un enfoque más orientado hacia la detección semántica de objetos, donde SAM podría identificar la presencia y ubicación de objetos de interés en la imagen sin necesidad de segmentar cada píxel individualmente. Esta aproximación se basó en la capacidad de SAM para dirigir su atención hacia regiones específicas de la imagen, determinadas por las cajas delimitadoras y puntos de referencia proporcionados como entrada.

Al emplear este enfoque, SAM adquirió la habilidad de comprender el contexto de la imagen mediante la identificación de características relevantes, lo que facilitó la detección precisa de objetos de interés.

En la figura 33 se puede ver como a partir de unas cajas escogidas con premeditación, SAM puede segmentar los elementos inscritos. En la parte 33b se puede ver la segmentación completa, en 33c y 33d SAM segmenta únicamente la llanta y el rin, por último en 33e se le puede decir a SAM que ignore el centro de la llanta (rin) y segmente solamente el neumático.

En este contexto, es fundamental abordar una cuestión clave: la selección de áreas de atención para SAM. Si bien en el ejemplo proporcionado se han seleccionado manualmente las regiones de interés, basándose en el conocimiento previo de la imagen del vehículo y sus características; esta aproximación no es sostenible ni práctica en un entorno real de detección de corales. En tales escenarios, carecemos de presunciones previas sobre la ubicación, forma y tamaño de los elementos de interés en las imágenes, es decir, las múltiples configuraciones de corales.

Para superar esta limitación, es necesario desarrollar un enfoque que permita la detección contextualizada de corales en las imágenes. Este enfoque implicaría la identificación automática de características que distinguen a los corales del entorno circundante, seguida de la creación de cajas delimitadoras

que encierren estas áreas de interés. Estas cajas delimitadoras actuarían como guías para SAM, restringiendo su atención y segmentación únicamente a los elementos identificados como corales dentro del entorno marino.

En particular, este enfoque requeriría la utilización de algoritmos de detección de características específicas de los corales, como su color, textura y morfología. Estos algoritmos deben basarse en técnicas de aprendizaje automático y procesamiento de imágenes para identificar patrones característicos asociados con los corales en diversas condiciones, incluidos corales completamente formados y corales blanqueados.

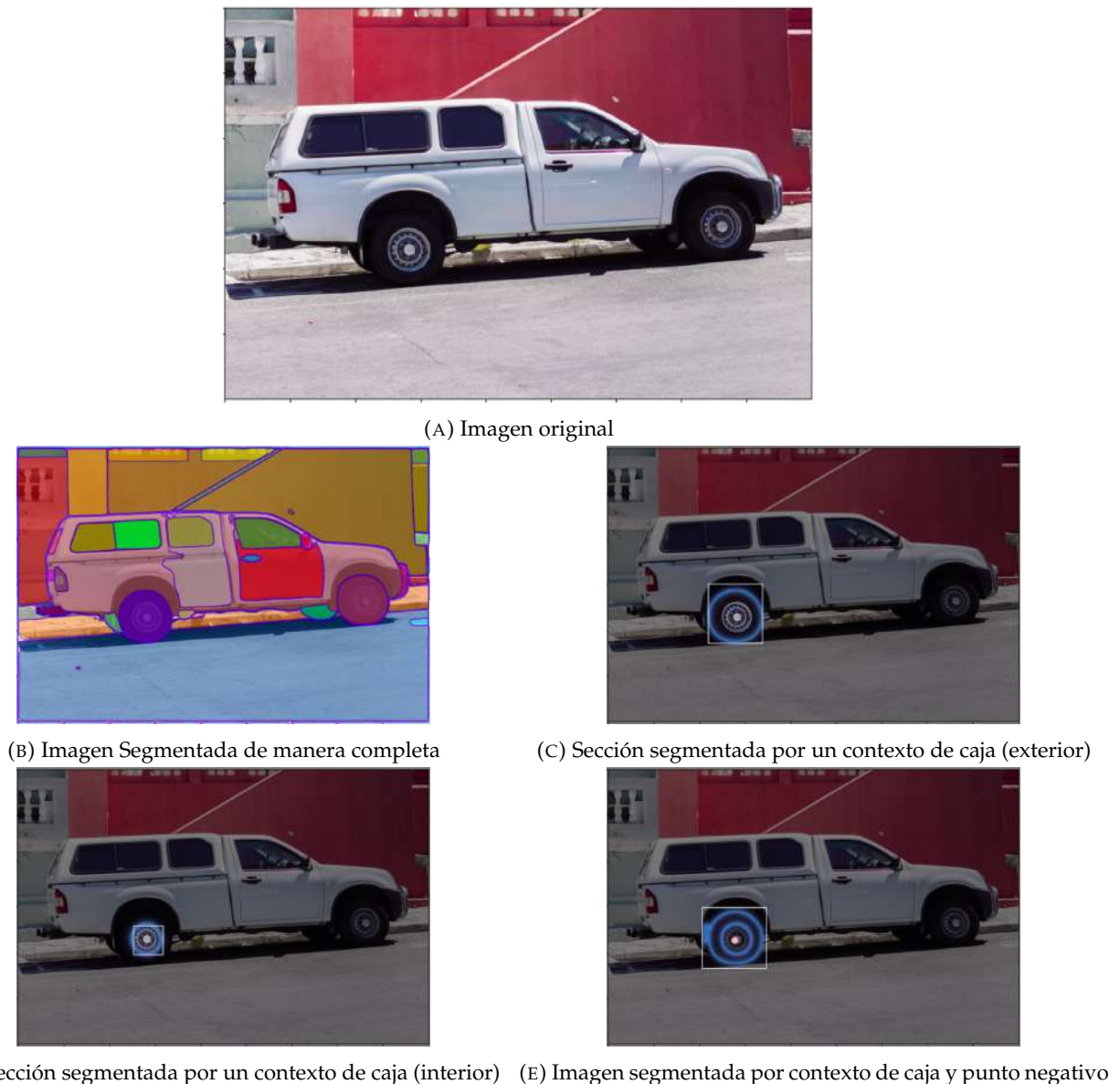


FIGURA 33: Ejemplo segmentación completa sin semántica y con parámetros por defecto

[4]

Para abordar la cuestión de la detección automática semántica de corales en imágenes marinas, se adoptó la utilización de la herramienta GroundingDino. Esta herramienta se caracteriza por recibir

como entrada frases simples o compuestas que describen los conceptos de interés, junto con ciertas operaciones de concatenación y/o priorización. Al integrar GroundingDino en la metodología, se establece una arquitectura de dos o más fases que facilita la detección efectiva de corales en las imágenes.

Es así que el proceso de detección se inicia mediante la formulación de frases que encapsulan las características distintivas de los corales y su entorno. Luego, mediante operaciones de concatenación y/o priorización, se refinan y estructuran estas frases para proporcionar una descripción más precisa y completa de los corales y sus características distintivas.

El uso de GroundingDino como parte de una arquitectura de múltiples fases permite una detección más precisa y contextualizada de corales en las imágenes marinas. Al integrar información semántica derivada de las frases de entrada con análisis de imágenes, se mejora la capacidad del sistema para identificar y segmentar eficazmente los corales, incluso en entornos donde la detección automática basada únicamente en características visuales puede ser desafiante. En última instancia, esta metodología ofrece una solución sólida y eficiente para la detección automática semántica de corales en imágenes marinas.

En la primera etapa, la IA de detección de objetos GroundingDino, se encarga de identificar la presencia de corales en la imagen. GroundingDINO escanea la imagen y crea múltiples cajas delimitadoras alrededor de los corales identificados. Estas cajas contienen información sobre las coordenadas espaciales y las dimensiones de cada coral detectado, lo que proporciona una indicación inicial de su ubicación y tamaño. Al especializarse en la identificación precisa de objetos en imágenes, la red examina la imagen en busca de características que correspondan a corales. Esto implica identificar formas, colores y texturas que son comunes en los corales y distinguirlas de otros elementos del entorno marino.

Una vez que GroundingDINO ha identificado los corales y generado las cajas delimitadoras, se procede a la segunda etapa del proceso. Aquí es donde entra en juego SAM, cuya función en esta etapa es llevar a cabo la segmentación precisa del contorno de los corales identificados en la etapa anterior. SAM analiza las cajas delimitadoras generadas por GroundingDino para determinar con precisión los bordes de cada coral.

Al segmentar el contorno de los corales, SAM crea máscaras de píxeles que indican las áreas exactas ocupadas por cada coral en la imagen. Estas máscaras permiten una representación visual detallada de la forma y la extensión de cada coral individual, lo que facilita su análisis y estudio posterior.

Una vez acopladas estas dos etapas, se realiza una prueba de inicio del modelo sin modificar sus parámetros, para observar su desempeño, detectando y segmentando los corales en primera instancia.

Se implementa un prompt inicial, cuya palabra clave de entrada en GroundingDino fue *coral*. Este prompt logró generar cajas de identificación sobre ciertos elementos presentes en las primeras imágenes de coral utilizadas en las pruebas. No obstante, la capacidad de detección de corales blanqueados se vio limitada debido a que, a pesar de ser corales, estos elementos no eran identificados correctamente en la imagen. Además, el umbral inicialmente seleccionado resultó inadecuado, ya que descartaba numerosas opciones que efectivamente correspondían a corales, incluyendo elementos periféricos o situados en el fondo, a veces borrosos, que no se encontraban en el plano principal de la captura.



FIGURA 34: Detección de objetos sin parametrizar, con el prompt "coral"

Debido a los parámetros iniciales de configuración utilizados, la etapa subsiguiente de SAM mostró deficiencias en la detección precisa de los contornos de los corales en la imagen. Esto se manifestó en la generación de cajas adicionales en la salida de GroundingDINO que no correspondían a corales. Por consiguiente, al emplearse estas cajas adicionales como entrada para el modelo SAM, este generaba máscaras adicionales basadas en dichas cajas, lo que conducía a errores en el proceso de segmentación final.

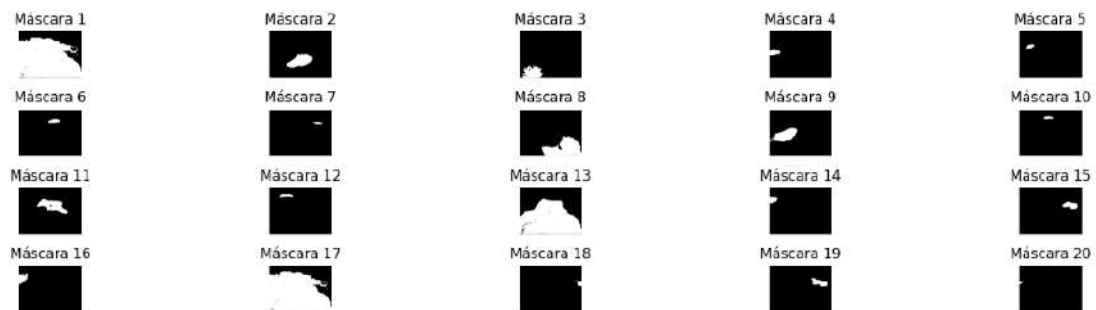


FIGURA 35: Máscaras resultantes de SAM con parametrización incorrecta

Considerando los resultados obtenidos, es evidente que cada modelo desempeña su función, aunque no de manera correcta, en la detección de corales. Por lo tanto, resulta imperativo identificar los parámetros fundamentales inherentes a cada modelo y ajustarlos adecuadamente para abordar eficazmente esta tarea específica. Este proceso de ajuste y optimización se abordará con mayor detalle en la sección subsiguiente.

7.3. Parametrización de los modelos

7.3.1. text prompt

Para el modelo GroundingDino, se llevaron a cabo pruebas utilizando diferentes prompts de texto con el fin de validar cuál podría funcionar de manera óptima. Inicialmente, se empleó el prompt "white". Los resultados obtenidos por el modelo de detección de objetos fueron efectivos en la identificación de regiones de coral blanqueado. Sin embargo, al aplicarlo a otras imágenes con contextos

distintos, como peces blancos o tanques de aire también blancos, el modelo identificaba incorrectamente estos elementos, lo cual dificultaba la distinción real entre coral blanqueado y otros objetos blancos en entornos marinos.

La segunda prueba consistió en utilizar el prompt "white . coral", indicándole a GroundingDino que concatenara ambos conceptos y buscara la intersección entre ellos. Esta estrategia representó una mejora significativa, ya que las cajas generadas por GroundingDino fueron más precisas en la detección de coral blanqueado. Sin embargo, esta aproximación descartaba los casos en los que el coral no estaba blanqueado, lo que podría afectar la precisión del cálculo del porcentaje de coral blanqueado respecto al coral sano.

Al utilizar únicamente el prompt "coral", se detectaron varios corales que la concatenación de conceptos no lograba identificar, lo que demuestra la robustez del modelo en la detección de corales en general. La pregunta que surge es: ¿cómo combinar estas dos formas de utilización?

La última prueba realizada consistió en el prompt "white coral . coral", con el objetivo de que el modelo de GroundingDino detectara tanto el coral blanqueado como el coral normal inicialmente, para luego en etapas posteriores realizar el cálculo correcto entre lo que es coral y lo que es coral blanqueado. Esta estrategia parece prometedora, ya que permitiría al modelo identificar tanto el coral blanqueado como el coral normal y luego realizar la distinción necesaria en etapas posteriores del procesamiento.

En las siguientes figuras se pueden ver algunos resultados parciales de la identificación de los corales y su segmentación con el prompt "white coral . coral":

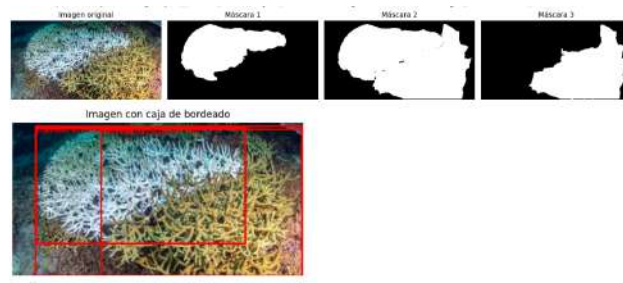


FIGURA 36: Coral con 1 sección de coral blanqueado y poco fondo

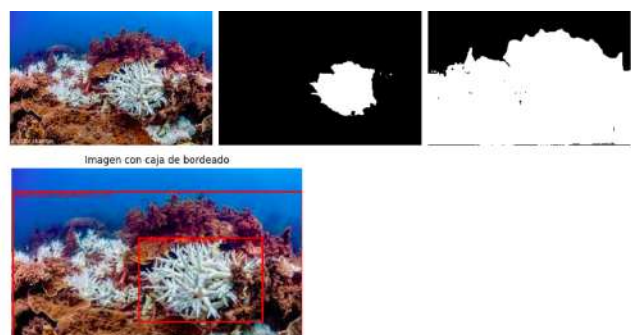


FIGURA 37: Coral en su mayoría sano y 6 regiones de diferentes tamaños blanqueadas

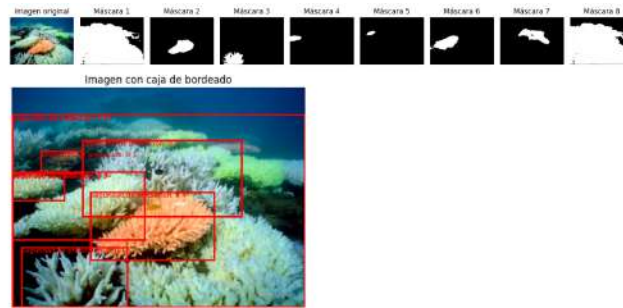


FIGURA 38: Varios corales individuales, muchos de los cuales están blanqueados

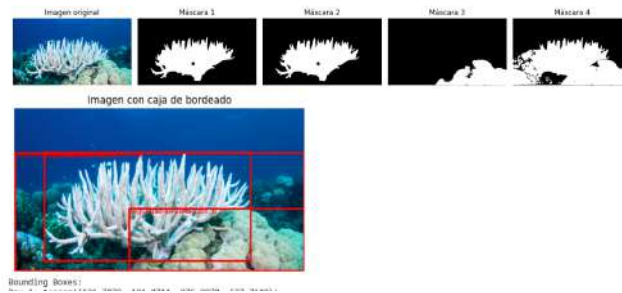


FIGURA 39: Dos corales diferentes pero blanqueados, fondo marino apreciable

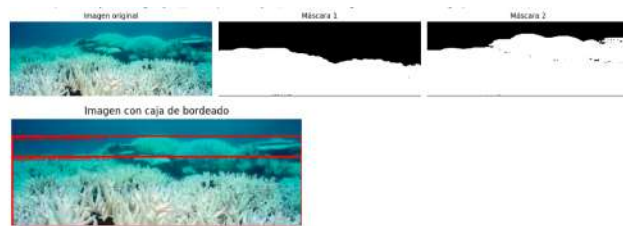


FIGURA 40: Casi toda la imagen tiene coral blanqueado

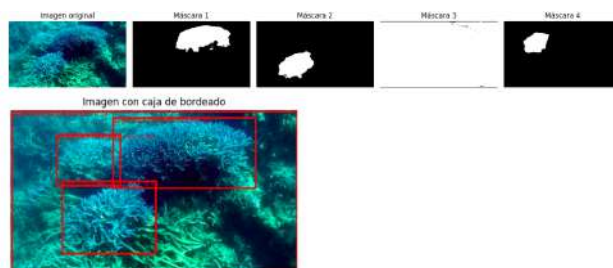


FIGURA 41: Imagen completa sin coral blanqueado

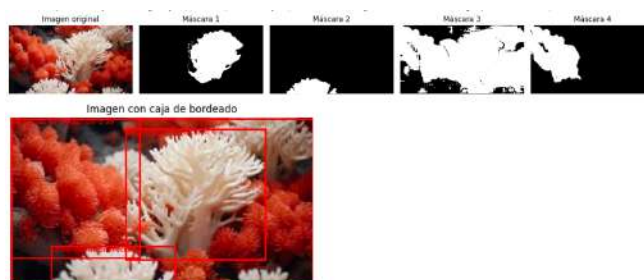


FIGURA 42: Coral rojo con 2 regiones idendificables de coral blanqueado



FIGURA 43: Coral blanqueado en primer plano, junto con otras partes de coral sano y bastante fondo marino adicional



FIGURA 44: Coral blanqueado en primer plano, junto con piedras y bastante fondo marino adicional

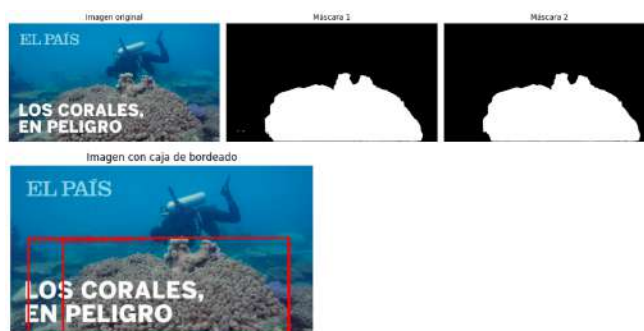


FIGURA 45: Coral sano y Letras blancas

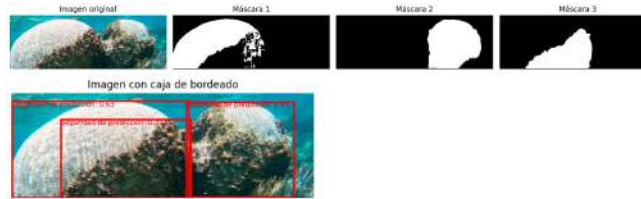


FIGURA 46: Coral blanqueado uniforme junto con coral sano

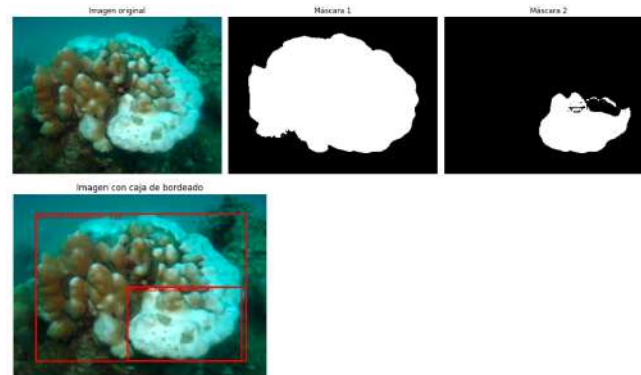


FIGURA 47: Coral mixto, partes blanqueadas y partes sanas

7.3.2. Umbral de detección para GroundingDino

GroundingDino presenta dos parámetros ajustables: un umbral para la detección de cajas (llamado `box_threshold`) y otro umbral para la interpretación del texto (`text_threshold`). Basándonos en las pruebas preliminares realizadas con el modelo, se pudo determinar que los valores óptimos generales para estos umbrales son de 0.25 y 0.3, respectivamente. Estos valores permitieron minimizar la cantidad de falsos positivos y maximizar la inclusión de verdaderos positivos relacionados con el coral y el coral blanqueado. En cuanto a la sensibilidad en la selección de las cajas, el valor también fue establecido mediante un proceso de prueba y error.

7.3.3. Variantes y Checksums de SAM

En la parametrización de SAM, se dispone de varios checkpoints que pueden ser utilizados. Durante las pruebas realizadas, se seleccionó el modelo más reciente hasta la fecha, que demostró una detección con la puntuación más alta en las pruebas respectivas, identificado como el checkpoint 4b8939. Además, el modelo SAM brinda la opción de escoger entre tres variantes según su tamaño y rendimiento: `sam_vit_h`, `sam_vit_l` y `sam_vit_b`, que corresponden a los modelos de alta, baja y base, respectivamente. Para las pruebas realizadas y el producto final entregado, se optó por emplear el modelo de alta precisión.

7.4. Detección de color (fase 3)

La detección de color desempeña un papel primordial en la arquitectura propuesta, ya que facilita la identificación de diferencias significativas en el color y las características regionales de los corales una vez que han sido segmentados. Este proceso es crucial para identificar posibles áreas de blanqueamiento coralino. En esta fase, se busca detectar incluso las variaciones mínimas de color en

la imagen, preferiblemente a nivel de discriminación de píxeles. Esto permitirá realizar un análisis de regiones de píxeles que no necesariamente son contiguas, para determinar la presencia de coral blanqueado frente a coral sano. Mediante este análisis, se pueden obtener porcentajes y proporciones precisas en relación con el coral en estudio.

Es fundamental que la detección de color sea capaz de reconocer similitudes entre elementos que son visualmente muy parecidos, considerando que algunas regiones pueden estar más iluminadas que otras, generando gradientes de color para el mismo elemento o región; también puede contener sombras proyectadas de otros elementos en la imagen e incluso el mismo coral, lo cual genera regiones más oscuras por defecto; por último a diferencia del aire, el agua no es totalmente traslúcida, por lo que siempre afecta el color verdadero de los elementos que rodea, especialmente cuando esos elementos están más lejos del lente de la cámara. Esto dificulta de manera especial la detección automática de lo que en teoría debería ser el mismo collar.

Además, se debe tener en cuenta que las regiones más distantes o pequeñas en la fotografía podrían tener una relevancia similar a las que se encuentran en primer plano; sin embargo, para este caso de uso no se contempla la perspectiva en la imagen, se parte del supuesto de que las fotografías se toman con el objetivo de abarcar la mayor superficie posible del coral en estudio sin perspectiva, dado que los corales tienden a tener dimensiones limitadas en longitud y profundidad y que al tomar las fotos se realizan a una distancia prudente para que en el área de la imagen contenga la mayor cantidad del coral en estudio.

7.4.1. Selección del algoritmo de discriminación de color

Después de realizar múltiples exploraciones de algoritmos y metodologías para la detección del color, se consideró inicialmente la detección por luminancia como uno de los primeros enfoques.

7.4.2. Luminancia

Esta aproximación heurística se fundamenta en la premisa de que el coral blanqueado puede exhibir un valor de luminancia considerablemente superior al de otros componentes de color presentes en los píxeles del coral en su estado normal. La detección se basaría en un simple criterio de selección, donde valores de luminancia por encima de un umbral preestablecido indicarían la presencia de coral blanqueado, mientras que aquellos por debajo representarían coral sano.

A pesar de que esta heurística inicial demostró un rendimiento satisfactorio en ciertas imágenes seleccionadas, presentaba serias limitaciones al intentar identificar coral en imágenes donde la luminancia del coral en su estado natural superaba el umbral establecido, especialmente cuando este se encontraba en primer plano, mientras que el coral blanqueado, situado más distante y, por ende, expuesto a una iluminación inferior, era detectado erróneamente como coral sano. Estas disparidades de luminosidad se atribuyen principalmente a la distribución de la luz proveniente de la linterna u otra fuente de iluminación utilizada para iluminar el fondo marino, la cual tiende a iluminar de manera más intensa los objetos en primer plano en comparación con aquellos más distantes.

Una de las principales dificultades del algoritmo radica en su dependencia de valores estáticos como criterio, donde el umbral debe ser predefinido. Bajo este enfoque, cualquier valor por encima del umbral se clasifica como blanqueado y por debajo como sano. Sin embargo, esta estrategia resulta problemática debido a la suposición de uniformidad en la iluminación de todas las imágenes, lo cual

no refleja la realidad. Algunas imágenes pueden capturarse en condiciones de iluminación deficiente, donde el coral puede estar distante o la potencia de la fuente de luz submarina podría ser insuficiente para una iluminación adecuada. Además, factores como la sobreexposición debido a la configuración de la cámara o la proximidad del coral a la superficie del mar en días soleados pueden alterar significativamente el balance de blancos. Por otra parte, ciertos tipos y especies de corales pueden no reflejar mucha luz naturalmente, ya sea estando sanos o enfermos.

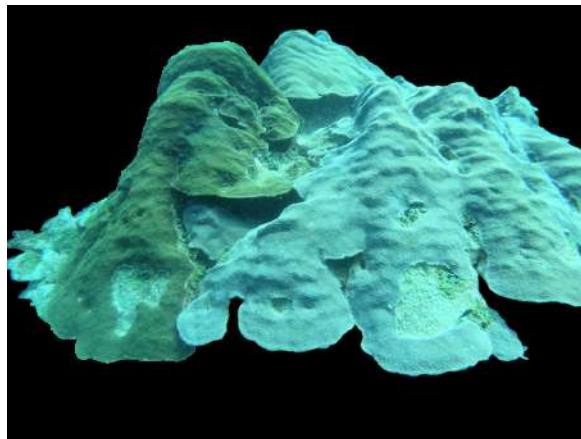
Una mejora provisional, frente a esta limitación, consiste en calcular el valor medio entre el punto más alto y más bajo de luminancia en el coral. Sin embargo, esta aproximación suele verse afectada por errores o zonas extremadamente iluminadas en el coral. Similar a los problemas asociados con los cálculos de percentiles en valores extremos, los valores promedio pueden estar sesgados por la presencia de valores extremos, lo que compromete la robustez del criterio de detección de corales blanqueados basados en este cambio.

En la Figura 48, se presenta la aplicación del algoritmo de luminancia para distinguir entre las regiones blanqueadas y las regiones con coral sano. En la subfigura 48a, se muestra un coral típico ubicado en el centro de la imagen, rodeado por el fondo marino. Tras procesar esta imagen con las fases 1 y 2 de la arquitectura propuesta, se obtiene la subfigura 48b, que logra segmentar adecuadamente el coral del fondo. Sin embargo, al aplicar directamente el algoritmo de luminancia con un umbral basado en el valor teórico medio en todo el rango dinámico ($l = 127$), se obtiene la subfigura 48c, donde no se logra una separación correcta entre las regiones de coral blanqueado y las de coral sano. Reducir significativamente el umbral, como se muestra en la subfigura 48d, empeora la situación.

Para abordar esta limitación, se implementó la mejora propuesta en el párrafo anterior, que consiste en calcular el umbral como el valor medio entre la luminancia más alta ($l = 255$) y la más baja ($l = 45$) en el coral. Esto resulta en un umbral de luminancia de 155, como se ilustra en la subfigura 48e. Sin embargo, tras una evaluación detallada, se determinó manualmente que el valor de luminancia que mejor separa el coral blanqueado del sano es 188, lo cual se refleja en la subfigura 48f. Este proceso de selección manual permitió obtener resultados más precisos en la detección de las regiones afectadas por el blanqueamiento coralino pero no se deriva de ninguna información a priori sino a posteriori del contenido de la imagen.



(A) Coral original con partes blanqueadas



(B) Coral segmentado por fase 1 y 2

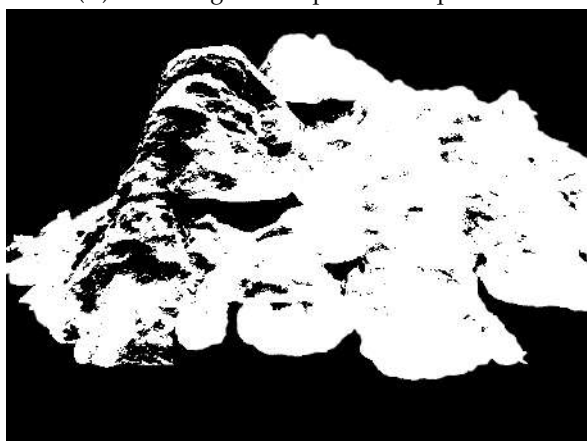
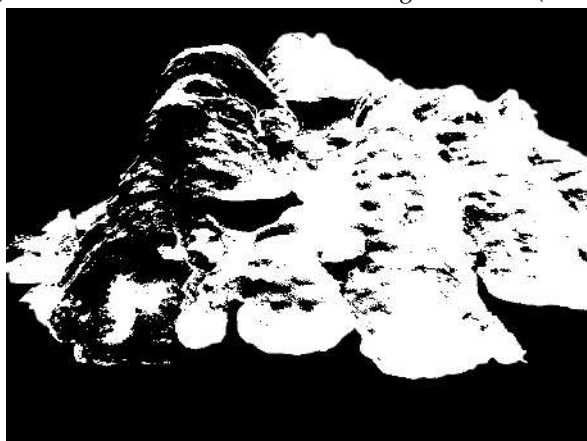
(C) Coral con umbral de la mitad del rango dinámico ($l = 127$)(D) Coral con umbral bajo ($l = 100$)(E) Coral con umbral del valor medio entre el valor más alto de luminancia y el mas bajo ($l = 150$)(F) Coral con umbral escogido a mano que aparenta ser el correcto para separar coral blanqueado del sano ($l = 188$)

FIGURA 48: Ejemplo de corte por luminancia, con diferentes valores para el mismo coral ya segmentado

Debido a esta falta de robustez del algoritmo de luminancia en tales escenarios, y considerando que numerosas imágenes se capturan en condiciones similares, este enfoque no proporciona un criterio confiable para la identificación del coral blanqueado en esta tercera etapa de análisis.

7.4.3. K-means

Al profundizar en la búsqueda de alternativas algorítmicas eficientes y de relativa facilidad de implementación, se optó por la utilización del algoritmo de agrupamiento K-means. Este enfoque se destacó por su capacidad para segmentar colores en grupos coherentes, lo que lo convierte en una herramienta prometedora para abordar la detección de color una vez ya segmentado el coral en etapas anteriores.

El K-Means clustering es un algoritmo de agrupamiento (clustering) utilizado en el aprendizaje no supervisado para dividir un conjunto de datos en grupos o clústeres basados en características similares.

El objetivo principal del algoritmo es particionar el conjunto de datos en K grupos, donde K es un número predefinido por el usuario, en este caso, 2 grupos: coral blanqueado y coral sano. [29]

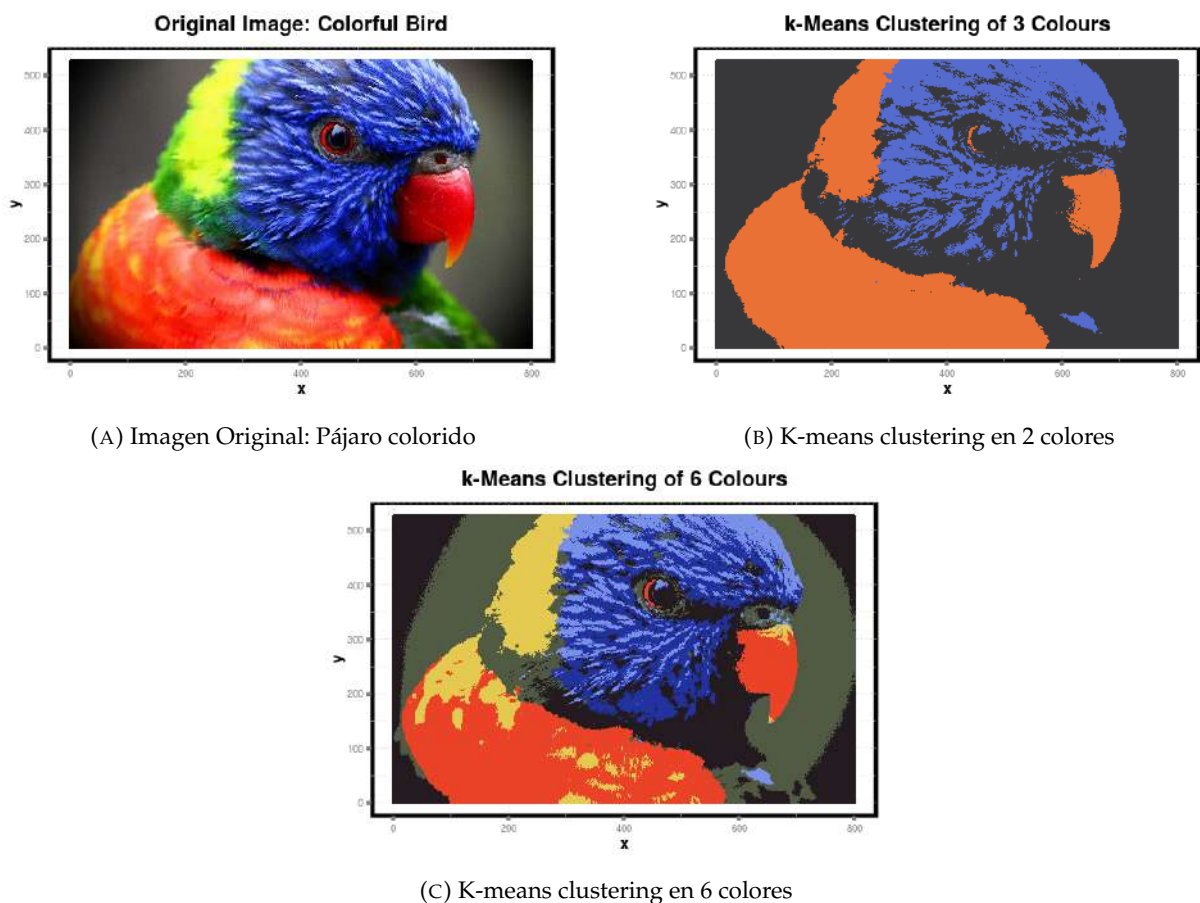


FIGURA 49: Ejemplo K-means clustering para una imagen de un pájaro colorido

Cada píxel de la imagen se considera como un punto en un espacio multidimensional, donde las dimensiones corresponden a los canales de color (por ejemplo, rojo, verde, azul en el espacio RGB). Para este caso, las características son simplemente los valores de los canales de color de cada píxel. Cada píxel se representa como un vector en el espacio de características. Luego se aplica el algoritmo

K-Means a los datos de píxeles, agrupando los píxeles en los 2 grupos basados en la similitud de sus valores de color. Después de que el algoritmo converge, cada centroide representa un color dominante en la imagen. Estos colores pueden ser utilizados en general para diversos fines, como la detección de la paleta de colores, pasterización, y en este caso, la extracción de las regiones de colores de la imagen.

Tras la generación de los clusters que representan los colores predominantes en la imagen, se procede a determinar cuál de estos clusters posee la mayor luminancia según los centroides hallados. Se parte del supuesto de que el coral blanqueado exhibirá la luminancia más alta, por lo que el centroide de mayor luminancia corresponderá a este. Posteriormente, mediante la extracción de la cantidad de píxeles correspondientes al coral blanqueado y al coral sano dada la división por clústeres, se calcula la proporción entre ambas divisiones. Este proceso permite obtener una estimación de la prevalencia relativa de coral blanqueado en comparación con el coral en su estado normal.

Se podría asumir que el algoritmo mencionado sería suficiente para segmentar cualquier imagen de coral en los colores deseados, logrando una adecuada distinción entre la proporción de coral blanqueado y coral sano; Sin embargo, es importante tener en cuenta que las imágenes extraídas típicamente contienen una variedad de elementos adicionales además del coral, tales como piedras, peces, arena, buceadores e incluso el propio fondo marino y el agua del océano. Por lo tanto, aplicar este algoritmo directamente a una imagen en bruto segmentará la imagen en colores similares, pero no necesariamente distinguirá específicamente el coral blanqueado en el área donde se encuentra el coral sano.

En la Figura 50, presentamos un ejemplo de esta situación, donde el algoritmo logra segmentar con precisión el degradado del fondo, pero no identifica de manera óptima la parte blanqueada del coral. Confundiendo la parte más clara del agua como coral blanqueado o otras partes más oscuras como coral sano, siendo estas realmente piedras y peces.



(A) Coral sano con partes blanqueadas, y fondo marino



(B) Imagen con k-means clustering aplicada a 3 colores

FIGURA 50: Ejemplo de clusterización de una imagen con el algoritmo k-means clustering

7.4.4. Parametrización k-means

La parametrización inicial del algoritmo k-means implica la definición de varios valores que guían su ejecución y determinan su criterio de convergencia. Dado que el problema que aborda es NP-hard, su ejecución puede prolongarse considerablemente para obtener una solución óptima. No obstante, la implementación utilizada ofrece resultados útiles en un tiempo computacional razonable.

Entre los parámetros del algoritmo se incluyen el número de iteraciones antes de concluir la ejecución de manera arbitraria y el epsilon, que representa la precisión mínima para finalizar el algoritmo. Para la implementación final, ambos criterios de terminación se mantienen activos, lo que significa que el algoritmo concluirá cuando se alcance el epsilon o cuando se completen el número predeterminado de iteraciones.

Tras extensivas pruebas, se determinó que un número óptimo de iteraciones era 10 y que el valor de epsilon era de 1, ambos en unidades absolutas dentro del rango dinámico 0-255.

7.4.5. Descripción detallada

Para la clusterización se utiliza el algoritmo de Lloyd, también conocido como el algoritmo de Lloyd-Max o simplemente el algoritmo de k-means, que es precisamente un método iterativo utilizado para realizar la agrupación de datos en k grupos distintos

1. El algoritmo comienza con una inicialización de k centroides en el espacio de características. Estos centroides pueden ser colocados inicialmente de diversas maneras, como al azar, mediante un muestreo uniforme, o basados en algún conocimiento previo del dominio de los datos.
2. Se calcula la distancia entre cada punto de datos y todos los centroides. Cada punto de datos es asignado al centroide más cercano, formando así k grupos iniciales.
3. Para cada grupo formado, se calcula el centroide, que es el punto medio de todos los puntos asignados a ese grupo. Este centroide se convierte en la nueva posición del centroide para ese grupo.
4. Los pasos 2 y 3 se repiten iterativamente hasta que se cumple un criterio de convergencia, como la estabilización de la posición de los centroides o la convergencia de la asignación de puntos a los grupos.

En cada iteración, el algoritmo de Lloyd actualiza los centroides para minimizar la distancia entre los puntos de datos y sus centroides respectivos, y reasigna los puntos a los centroides más cercanos. Este proceso continúa hasta que se alcanza un estado de estabilidad en el que los centroides y las asignaciones de puntos permanecen relativamente constantes basados en los criterios de finalización expuestos anteriormente. [28]

7.4.6. Centroides aleatorios

En las etapas iniciales, se exploró la selección aleatoria de centroides como método para el primer paso del algoritmo. Aunque esta aproximación permitía un inicio rápido sin prejuicios sobre la imagen, la segmentación de color resultante, aunque útil, no alcanzaba su máximo potencial. Interacciones adicionales revelaron que esta condición de inicio aleatorio, si bien fácil de implementar, no favorecía la rápida convergencia del algoritmo. En condiciones especiales, como aquellas de sombra o distantes, se observó que ciertas partes del coral no se asignaban correctamente al cluster correspondiente, a pesar de la ejecución de múltiples iteraciones en un intento por mejorar los resultados. Además, en ocasiones, incluso tras llevar a las 10 iteraciones elegidas al final, no se lograba una segmentación adecuada para imágenes particulares que presentaban características muy específicas. Este desafío se vio agravado por la limitación de utilizar únicamente tres clusters para abarcar toda la imagen, la cual presentaba una amplia gama de valores de píxeles, así como las cinco dimensiones en consideración: ancho, alto y los tres componentes de color (RGB) de cada píxel.

7.4.7. Centroides basados en Kmeans++

Una aproximación más efectiva se encontró en el uso del algoritmo k-means++, una técnica diseñada para mejorar la inicialización de los centroides en el algoritmo k-means. Este método, comúnmente empleado en análisis de agrupamiento de datos, busca mitigar los efectos adversos de la inicialización aleatoria o determinística de los centroides en la calidad y convergencia de la agrupación resultante de color. El proceso de inicialización de k-means++ se lleva a cabo en varias etapas iterativas, comenzando con la selección aleatoria de un centroide inicial y procediendo a la selección ponderada de centroides subsiguientes basados en las distancias calculadas.

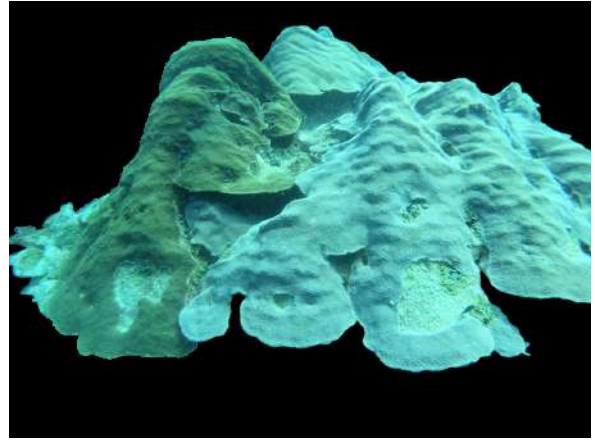
1. Se selecciona inicialmente un centroide al azar de entre los puntos de datos disponibles. Esta elección aleatoria establece el primer centroide y sirve como punto de partida para el proceso de inicialización.
2. Para determinar los centroides restantes, se calcula la distancia de cada punto de datos al centroide más cercano que ya ha sido seleccionado. Esta distancia se utiliza para calcular una distribución ponderada, donde los puntos más alejados tienen una probabilidad proporcionalmente mayor de ser seleccionados como centroides subsiguientes.
3. Se elige un nuevo centroide utilizando la distribución ponderada obtenida en el paso anterior. Este proceso se repite hasta que se seleccionan todos los centroides requeridos para el algoritmo k-means. [8]

k-means++ proporcionó una estrategia sistemática y eficaz para la inicialización de centroides en el algoritmo k-means, lo que mejoró significativamente la calidad de las detecciones de color resultantes. Cuando se aplicó a la cuantización de colores en las imágenes del banco creado, k-means++ permitió reducir el número de iteraciones del algoritmo para una imagen particular mientras se conserva una representación visualmente significativa de los clústeres escogidos para las dos condiciones de coral blanqueado y sano. Las pruebas normalmente dieron una mejora de hasta el 20 % en la calidad de las máscaras generadas vs. la selección de los centroides por métodos aleatorios.

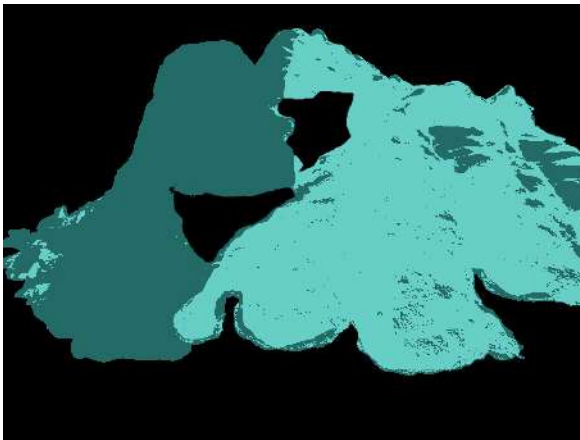
En la figura 51, se presenta una comparativa entre dos métodos de inicialización de clusters: uno basado en una selección aleatoria (figura 51c) y el otro generado mediante la inicialización por k-means++ (figura 51d). Se observa que, particularmente en los bordes o en las transiciones abruptas de color, el algoritmo de inicialización aleatoria exhibe limitaciones en la detección precisa de los colores. En contraste, el método de inicialización por k-means++ tiende a explorar con mayor detenimiento las áreas de sombra y los contornos ligeramente más oscuros, los cuales están asociados a un coral blanqueado.



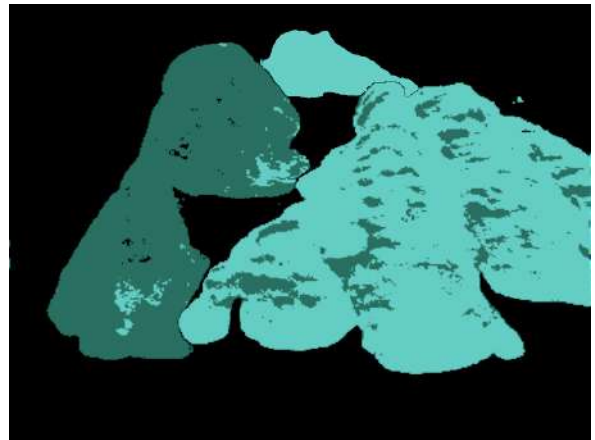
(A) Coral original con partes blanqueadas



(B) Coral segmentado por fase 1 y 2



(C) Coral con colores cuantizados mediante inicialización aleatoria



(D) Coral con colores cuantizados mediante inicialización por kmeans++

FIGURA 51: Ejemplo de cuantización de color, por los diferentes métodos de inicialización para k-means

8. Resultados

Las fotografías subacuáticas previamente seleccionadas del banco de imágenes fueron introducidas en el modelo desarrollado para su análisis. A partir de este proceso, se generaron los resultados correspondientes al porcentaje de blanqueamiento de los corales, tanto aquellos segmentados manualmente como los segmentados por el modelo. Además, se calculó el porcentaje de error relativo entre ambos conjuntos de resultados. Las fórmulas utilizadas para estos cálculos se expresan de la siguiente manera:

$$\%Blanqueamiento = \frac{A_{blanqueada}}{A_{total}} * 100 \% \quad (6)$$

Donde:

$A_{blanqueada}$ = Número de pixeles del área blanqueada.

A_{total} = Número de pixeles del área total del coral.


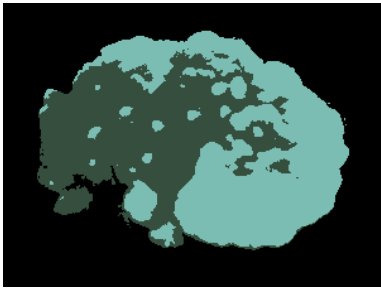
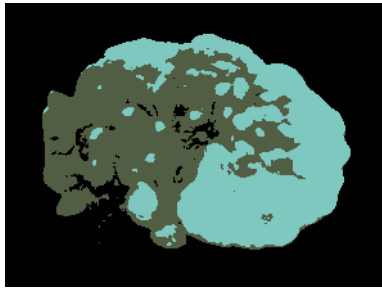

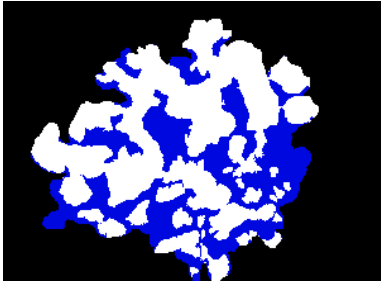
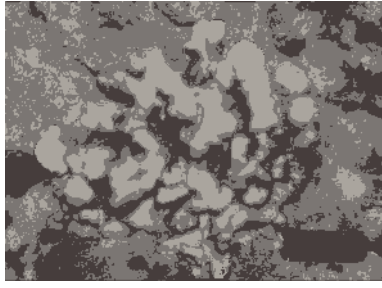

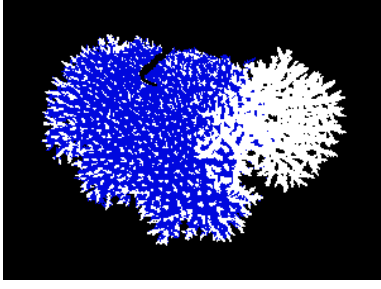
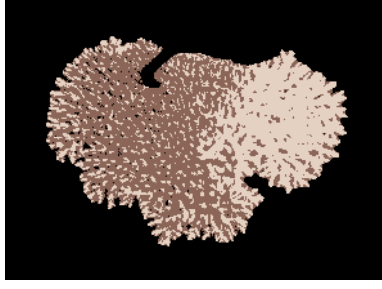




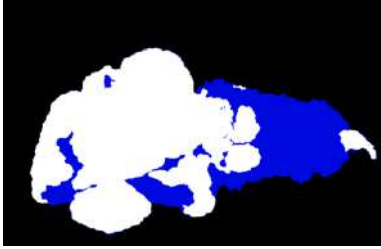
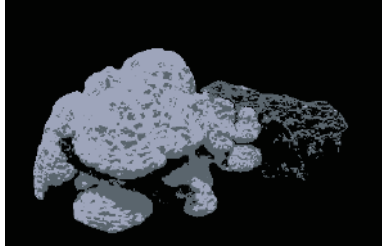
$$\%Error = \frac{|B_0 - B|}{B} * 100 \% \quad (7)$$

Donde:

B_0 = Porcentaje blanqueamiento obtenido manualmente.

B = Porcentaje blanqueamiento obtenido del modelo.

Además, para una mejor comprensión de los resultados obtenidos, se procedió a visualizar gráficamente el desempeño del modelo. En la tabla 1, se muestran la fotografía original, la imagen del coral segmentado manualmente y la imagen del coral segmentado por el modelo. Sin embargo, debido a limitaciones de espacio, se presentará en la tabla de imágenes resultantes una muestra de cinco fotografías seleccionadas del banco total. Es importante destacar que en la tabla de resultados 2, se incluyen los datos correspondientes a todas las imágenes procesadas.

#	Imagen original	Segmentación manual	Segmentación modelo
1			
2			
3			
4			
5			

CUADRO 1: Tabla con las imágenes originales, la segmentación manual y la resultante del modelo

Resultados del modelo			
#Imagen	% Blanqueamiento del experto	% Blanqueamiento del modelo	% Error
1	52.02	47.60	9.29
2	61.06	32.26	89.27
3	45.02	45.28	0.56
4	74.76	48.71	53.49
5	68.53	57.06	20.10
6	26.44	27.48	3.77
7	73.33	45.32	61.79
8	13.62	28.95	52.97
9	12.59	19.03	33.86
10	100.00	83.85	19.26
11	40.30	35.79	12.61
12	100.00	53.82	85.82
13	33.94	45.56	25.50
14	100.00	73.32	36.38
15	69.56	69.55	0.01
16	82.42	62.49	31.90
17	37.44	42.44	11.78
18	37.09	54.53	31.98
19	87.79	100.00	12.21
20	68.67	43.37	58.35
21	20.38	48.61	58.06
22	100.00	100.00	0.00
23	30.85	42.34	27.13
24	100.00	14.90	571.04
25	39.93	43.37	7.94
26	100.00	36.71	172.38
27	6.47	49.86	87.03
28	0.71	66.89	98.94
29	66.38	57.91	14.63
30	0.91	41.67	97.81
31	0.82	37.99	97.84
32	9.72	47.98	79.75
33	100.00	43.92	127.70
34	100.00	13.76	626.74
35	100.00	53.04	88.53
36	100.00	27.40	264.90
37	100.00	56.87	75.83


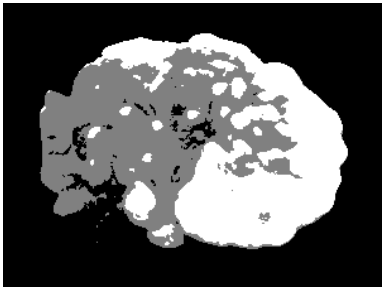
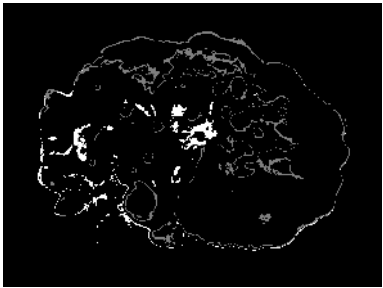
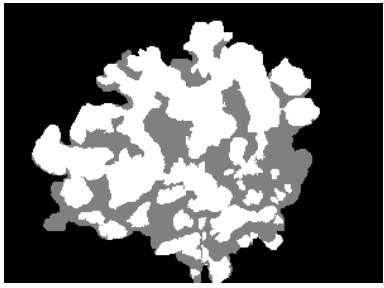
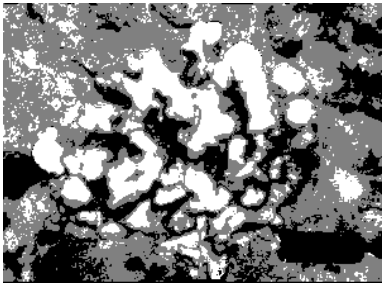
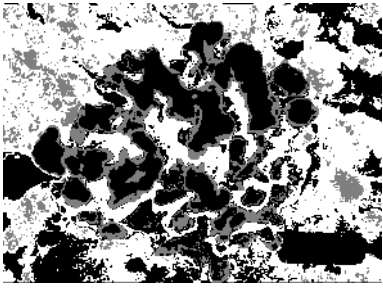
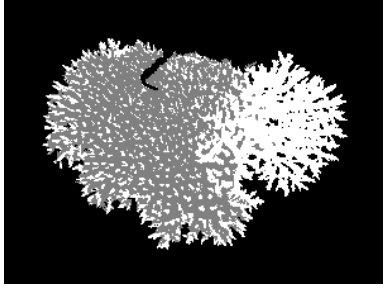
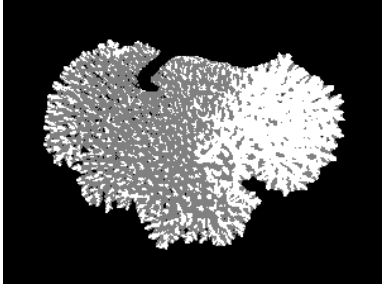
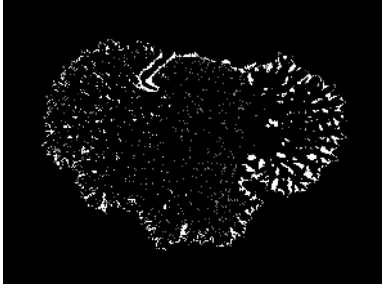

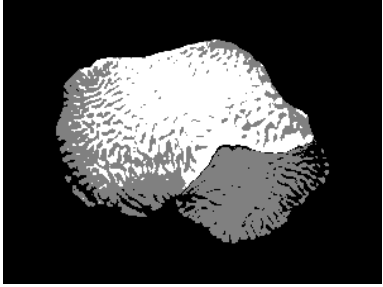

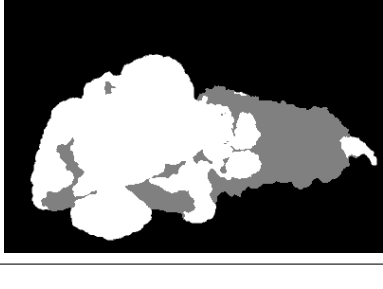
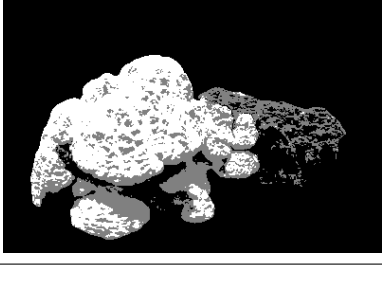
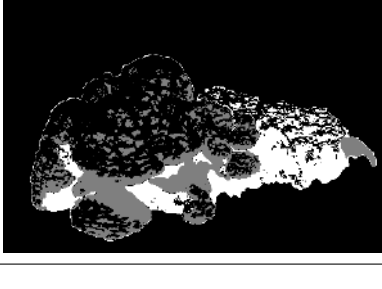
CUADRO 2: Tabla de resultados y error del modelo

Los resultados obtenidos revelan en su mayoría, que los porcentajes de error más bajos se observaron en las fotografías donde el coral fue capturado con una iluminación directa, minimizando la presencia de sombras sobre el coral y se ven más definidos en la imagen. Este fenómeno se evidencia en la muestra de la tabla 1, donde se destaca el coral número 1 y número 3 con un error del 9.29 %

y 0.56 % respectivamente, por otra parte, se evidencia que hay casos en los que se alcanzó un error del 89.27 % y 53.49 % para los corales 2 y 4 respectivamente, donde en el caso del coral 2 no era muy evidente diferenciar el área que se estimó como coral y fondo, el modelo tomó toda la imagen como coral, sin embargo es importante mencionar, que al discutir estos resultados con el experto, nos dijo que para este caso puede que lo que tomamos como fondo es realmente ambiguo porque al no tener un plano más de lejos, puede que esa región sea coral como puede que no lo sea, por lo tanto puede que el modelo que detecta coral no se haya equivocado en realidad. Lo mismo sucedió en el caso del coral 4, ya que no fue muy claro para el experto si en realidad toda el área que tomó como blanqueada en realidad estaba toda blanqueada u lo que se ve amarillento es producto de sombras y suciedad, o si estas zonas amarillentas son que aún no están blanqueadas.

Esta situación de ambigüedad, de acuerdo a la estimación del experto, se presentó en varias imágenes de la base de datos ya sea ambigüedades de si es coral o fondo y si cierta región del coral está blanqueada o no; así que en estos casos, el experto optó por tomar una decisión de lo que mejor consideró. En otros casos, como se mencionó anteriormente, la presencia de muchas sombras o zonas muy iluminadas hizo que el algoritmo K-means tuviera confusiones en la clasificación de fondo, coral sano y coral blanqueado y dar una correcta cuantización de color.

Por otro lado, se observa otro escenario en el que se evidenciaron los errores más altos de todo el conjunto de imágenes con porcentajes de error que superan el 100 % e incluso el 500 %, al revisar las imágenes correspondientes, lo que tenían en común es que fueron las imágenes que no presentaban blanqueamiento, y por lo tanto, estaban completamente sanos, así que al momento de llegar a la fase 3 de K-means, toda región que se observaba más iluminada que el resto lo tomó como zona blanqueada cuando la máscara manual de referencia cuenta con un blanqueamiento de 0 %, como consecuencia, genera un error demasiado alto.

#	Imagen original	Segmentación experto	Diferencias
1			
2			
3			
4			
5			

CUADRO 3: Tabla de segmentación hecha por el experto, la resultante del modelo y las diferencias en las imágenes pixel a pixel

A pesar de que el error relativo muestra cifras significativas en algunas imágenes, se tomó la decisión de llevar a cabo una verificación de similitud entre ellas mediante la comparación de diferencias. Esto permitió establecer un criterio más sólido para comparar los resultados. En este sentido, se procedió a normalizar ambas imágenes en color y a comparar las diferencias píxel a píxel entre ellas. Este proceso se detalla en la tabla 3, donde la primera columna representa la imagen segmentada por el experto, la segunda columna corresponde a la imagen generada por el modelo, y la tercera columna muestra las discrepancias píxel a píxel entre las dos imágenes normalizadas.


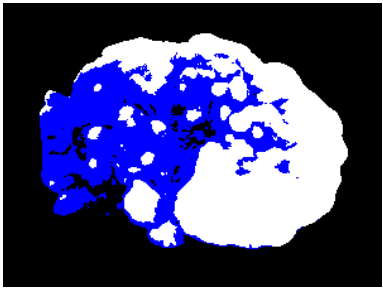
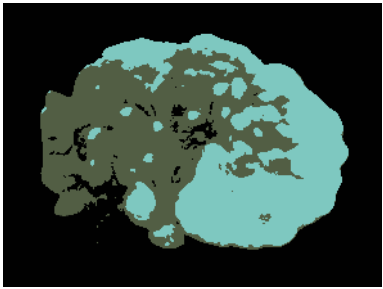

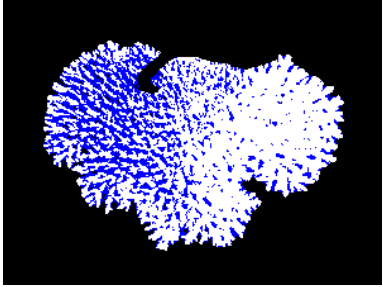
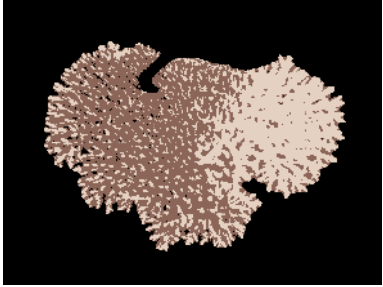

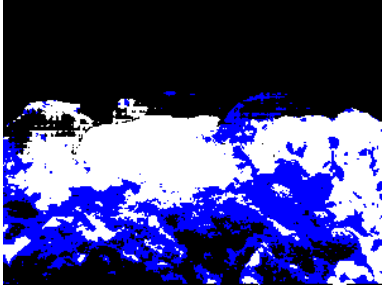
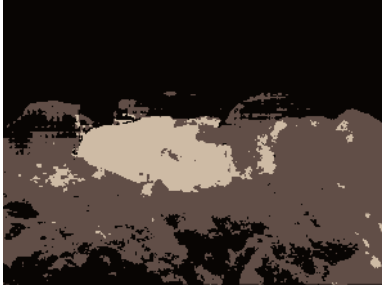
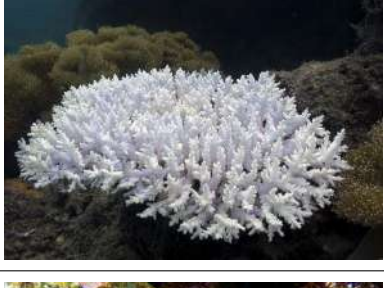
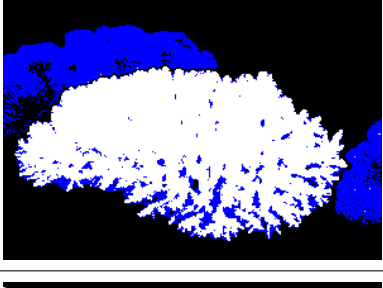
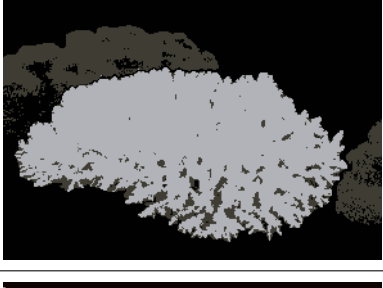

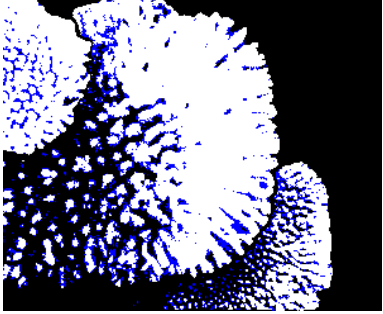
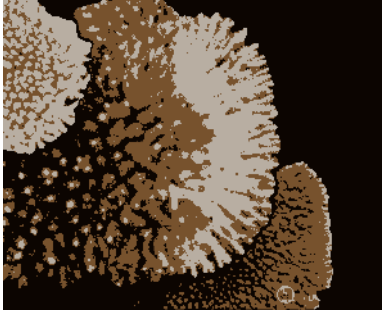
El análisis revela que, en general, las diferencias entre las imágenes no son significativas. Las mayores disparidades se observan principalmente en las regiones de los bordes y en aquellas áreas donde la presencia de sombras dificulta la detección.

En la literatura revisada, no se ha identificado ninguna investigación previa que haya abordado específicamente la tarea de detección y segmentación de corales, así como la estimación cuantitativa de su grado de blanqueamiento usando redes neuronales artificiales o alguna otra técnica diferente a la que usamos para la segmentación manual. Debido a esto, la comparación directa de los resultados obtenidos mediante la arquitectura desarrollada en este proyecto con estudios anteriores no resulta factible debido a la ausencia de trabajos previos que aborden de manera similar esta problemática.

En la tabla 4 se presenta una comparación entre diferentes métodos de cuantización de imágenes aplicados a la detección de corales blanqueados. En la primera columna se exhibe la imagen original, seguida por la imagen cuantizada utilizando el algoritmo de luminancia en la segunda columna, y la imagen cuantizada utilizando el algoritmo k-means después de la fase 3 del modelo en la tercera columna. Aunque se aprecia una sutil diferencia en algunas filas, como en la fila 1 y la 4, en general se observa que el algoritmo de luminancia no logra discernir adecuadamente entre el coral blanqueado y el sano, generando numerosos falsos positivos de color blanco que no corresponden ni siquiera a corales similares al blanqueado.

Por ejemplo, en la fila 3 se identifican como corales blanqueados a corales que no son blancos, sino de un tono pastel similar al coral blanqueado presente en el centro de la imagen. Mientras tanto, el algoritmo de cambios es capaz de detectar estas diferencias y seleccionar únicamente el coral blanqueado. Por otro lado, en la fila 5, se observa que el algoritmo de luminancia incluso identifica como coral blanqueado áreas que son completamente sanas y de color rojo en la esquina inferior derecha, debido a su luminosidad y vivacidad, lo que genera una confusión irónica en el algoritmo.

En contraste, en la parte central de la imagen, donde si hay una combinación de corales sanos y blanqueados, el algoritmo de luminancia identifica erróneamente elementos del coral sano como blanqueado en algunos puntos. Esta falta de robustez en la implementación de luminancia llevó a optar por el método de k-means en la última etapa, el cual ofreció resultados superiores, siempre y cuando la máscara de segmentación se detectara correctamente.

#	Imagen original	Cuantización luminancia	Cuantización k-means
1			
2			
3			
4			
5			

CUADRO 4: Tabla con ejemplos de imágenes originales, la cuantización en fase 3 por luminancia y la cuantización en fase 3 por k-means

9. Conclusiones

La detección, segmentación e identificación de la parte blanqueada de un coral es intrínsecamente un desafío complejo. Requiere un procesamiento detallado en secciones debido a que no se trata únicamente de identificar objetos como se ha hecho tradicionalmente, sino de discernir una cualidad específica, en este caso, el color, de un elemento muy particular como el coral. Por ende, resulta conveniente haber creado una arquitectura que integrara lo mejor de diversas redes neuronales disponibles en la actualidad, incluyendo modelos de detección de objetos, segmentación de formas y cuantización del color. A pesar de ello, es necesario realizar una calibración fina de múltiples parámetros para optimizar la eficacia en la consecución de los objetivos.

En la última fase del proceso, que comprende la cuantización y detección del color, es crucial tener en cuenta las partes más y menos iluminadas de una imagen, que pueden incluir sombras, degradados en la iluminación de los objetos e incluso impurezas propias del objeto, todos los cuales pueden alterar ligeramente la percepción del color. Si bien esto se puede minimizar calibrando adecuadamente las etapas anteriores para limitar o mitigar la consideración de estos elementos en etapas posteriores, siempre quedan algunos elementos y casos particulares que escapan a nuestro control y que no pueden ser filtrados antes de llegar a la última fase.

A pesar de que la aproximación al problema se basó en datos empíricos obtenidos a partir de un criterio experto, este enfoque sigue siendo en gran medida heurístico y, por lo tanto, no garantiza una comprensión precisa de lo que constituye un coral blanqueado y qué constituye un coral sano. Aunque el experto tiene una experiencia abundante en el campo que le permite entender esto, trasladar esa experiencia a la detección precisa de los píxeles para las máscaras reales resulta en una fidelidad limitada debido a la naturaleza difusa del criterio de detección de coral blanqueado. Este proceso también está sujeto a un margen de error humano adicional debido a detalles como sombras, elementos externos y elementos ambiguos en las imágenes, además de la resolución en píxeles de las imágenes que no favorece esta tarea.

Aunque inicialmente se creyó que las imágenes marinas, con su paleta de colores predominantemente azules, podrían afectar negativamente la detección de objetos y, por ende, la detección de corales blanqueados que presentan un tono azulado, la realidad demostró lo contrario. Las redes neuronales entrenadas con una amplia variedad de imágenes, formas e incluso paletas de colores, desarrollaron un criterio robusto en términos de colores y no se vieron significativamente afectadas al detectar las formas y contornos necesarios para la detección de corales, independientemente del entorno.

La decisión tecnológica de utilizar Python como lenguaje de referencia y de implementación para la arquitectura fue acertada. Esto permitió la exploración y experimentación con múltiples modelos de manera integrada en un flujo común, lo que facilitó la rápida eliminación de enfoques que no agregaban valor, como la detección por luminancia, modelos de segmentación de objetos no tan robustos y entrenamientos no muy resilientes a condiciones adversas.

Dado que los investigadores de corales marinos deben estimar visualmente el porcentaje de blanqueamiento de los corales, incluso de manera manual en editores de imágenes, la implementación de un modelo de redes neuronales que pueda realizar esta tarea representa una gran ventaja para ellos. Dado el gran volumen y la variedad de fotografías submarinas que deben capturarse y analizarse



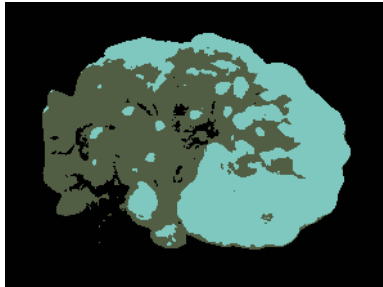
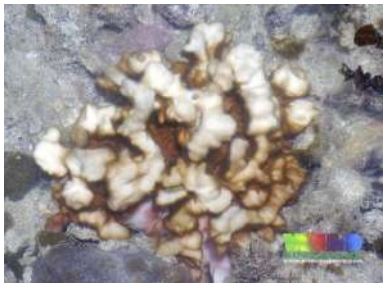
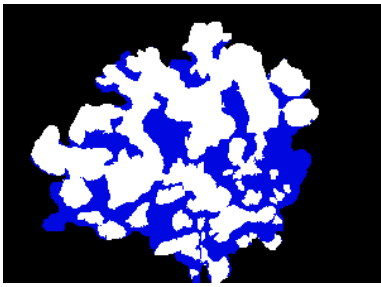
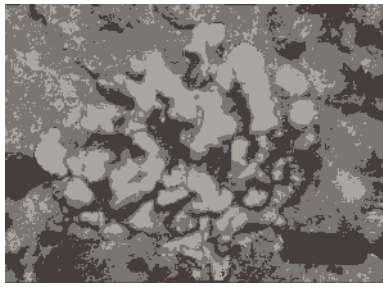

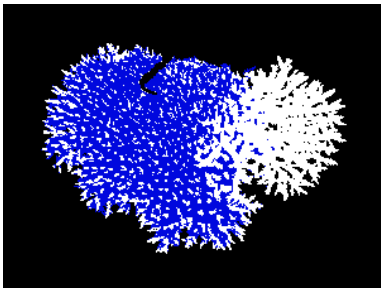
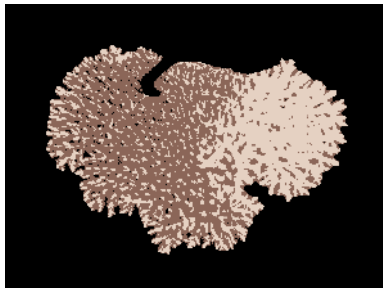


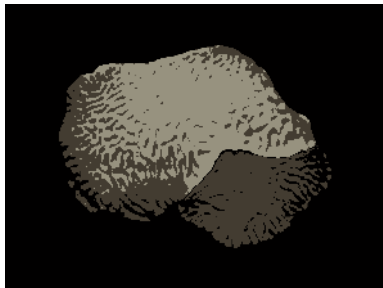

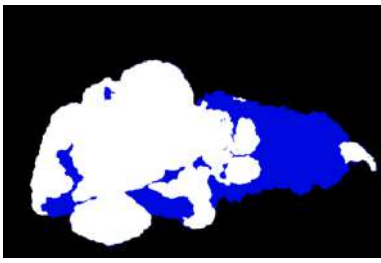
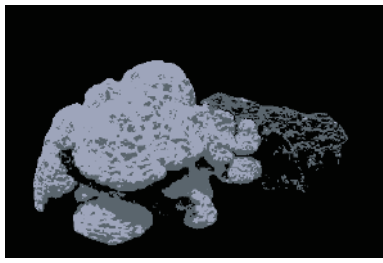
en un arrecife durante diferentes ángulos y períodos de tiempo, la carga de trabajo asociada con el análisis visual de cada fotografía es considerable.


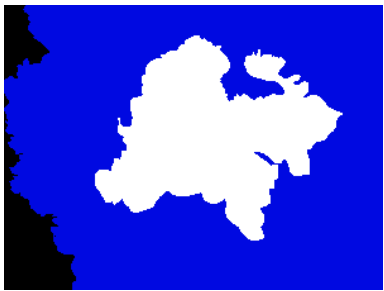
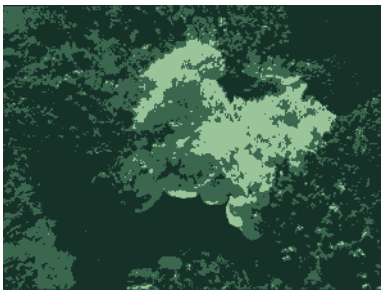




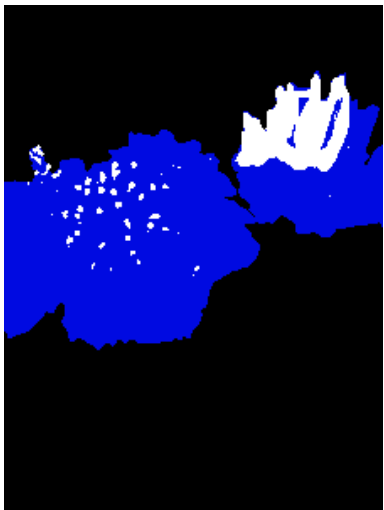







Durante el proceso de búsqueda de la literatura científica pertinente, no se identificaron estudios previos que hayan abordado de manera específica la tarea integral de detección, segmentación de corales y estimación cuantitativa de su blanqueamiento con redes neuronales, tal como se ha propuesto en el presente proyecto de investigación. La ausencia de investigaciones previas que aborden esta problemática particular impide la realización de comparaciones entre los resultados obtenidos en este estudio y los hallazgos de trabajos anteriores. Este vacío en la literatura subraya la relevancia del enfoque adoptado en la presente investigación, así como la necesidad de contribuir a la aplicación de modelos de redes neuronales en el campo de la detección y análisis de corales en entornos submarinos, siendo este un primer paso para abordar una brecha en el uso de redes neuronales, para brindar herramientas que de cierta forma contribuyan en este campo de la salud de los arrecifes de coral y su respuesta al cambio climático y otros factores ambientales.


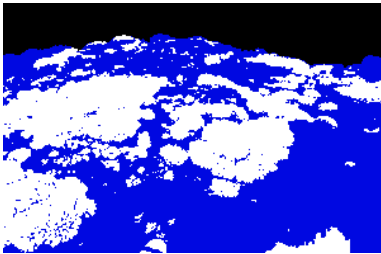
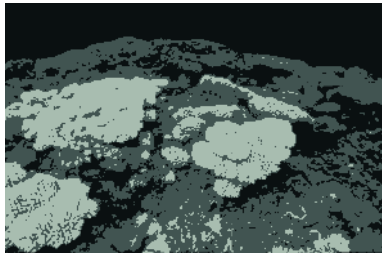


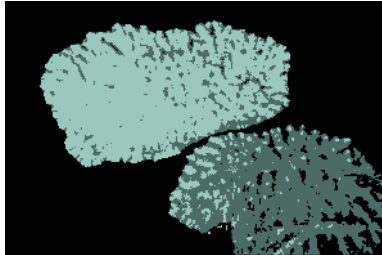

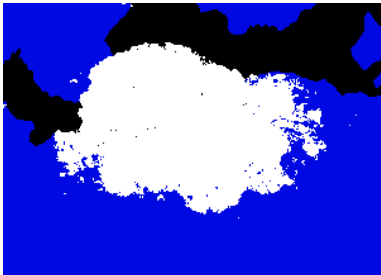
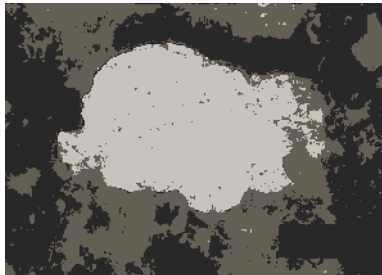


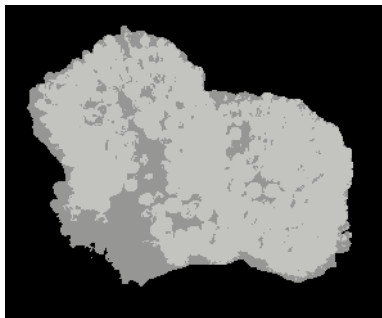

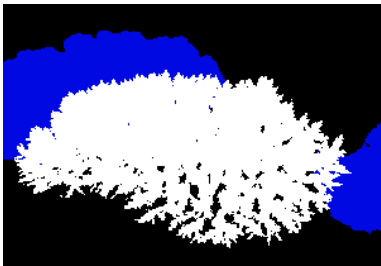
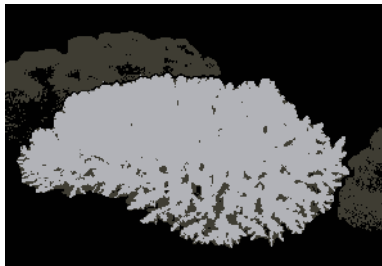
Un punto crucial observado durante este proyecto es que el porcentaje de blanqueamiento de cada coral depende en gran medida del ángulo de captura de la fotografía, ya que sólo se podrá estimar el blanqueamiento únicamente del plano que se está observando, sin saber a ciencia cierta si desde otros ángulos se aprecian más regiones blanqueadas. Esta observación sugiere que, para trabajos futuros, se podría mejorar la resolución de este problema mediante la captura de varias fotos desde diversos ángulos por coral, o de preferencia, el uso de videos en lugar de fotografías, lo que permitiría obtener una estimación más precisa del porcentaje de blanqueamiento desde múltiples ángulos y los modelos utilizados también pueden ser implementados para aplicaciones de video.


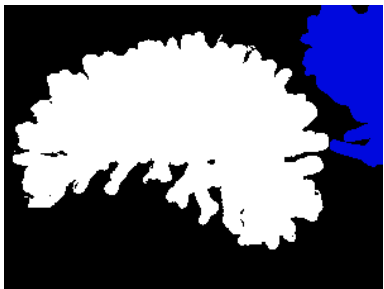


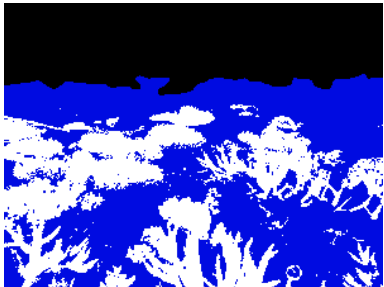
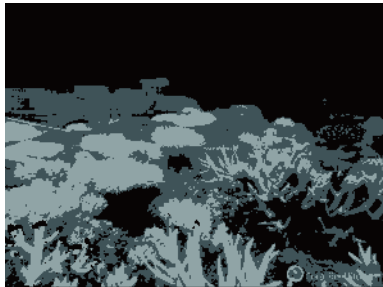


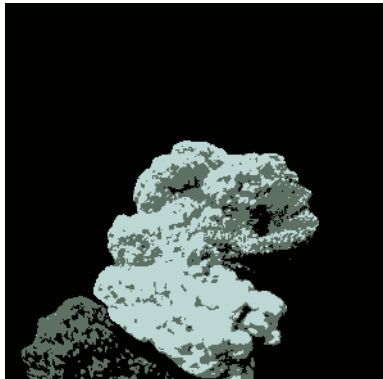

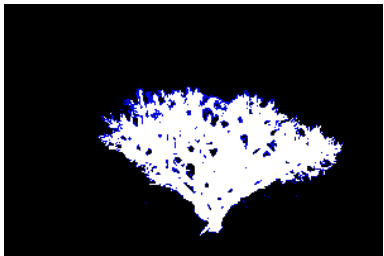


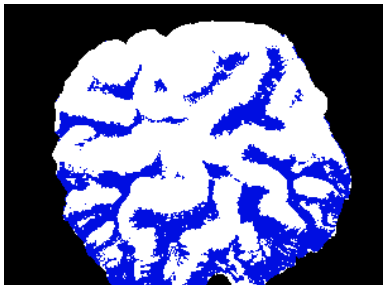

Por otro lado, se plantea como un desafío adicional la mejora de la segmentación del color, para lo cual se sugiere la exploración de algoritmos alternativos que potencialmente arrojen resultados superiores. Específicamente, se requiere abordar la problemática de las sombras, así como la presencia de corales con tonalidades muy claras similares al blanco, que comprometen el desempeño del algoritmo de K-means.


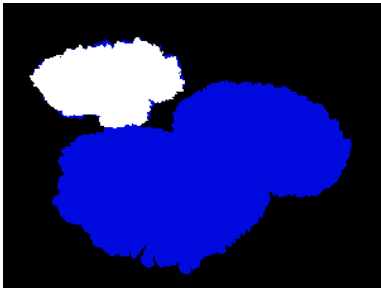
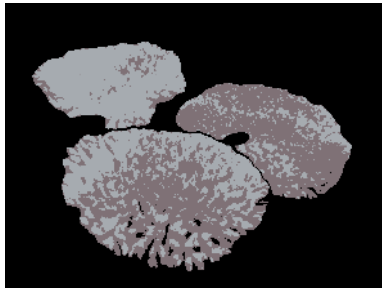

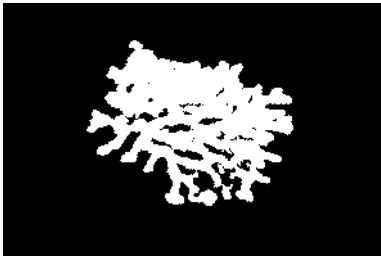


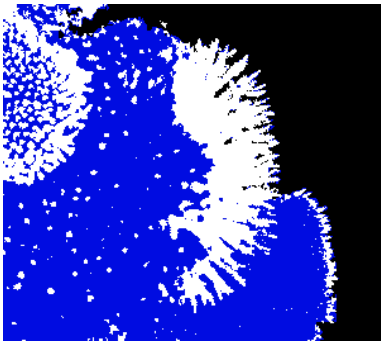
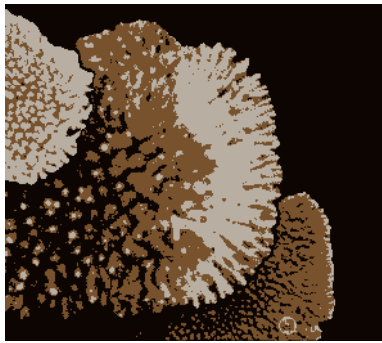




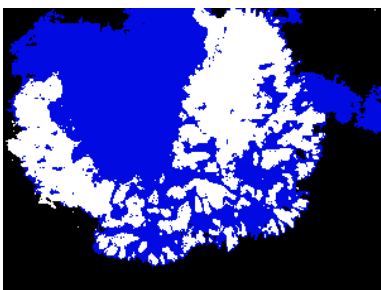
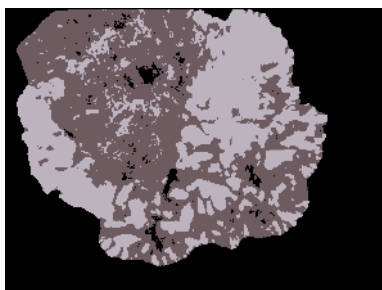
10. Anexos


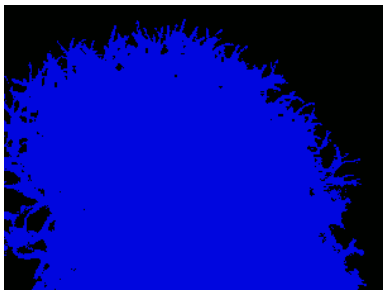
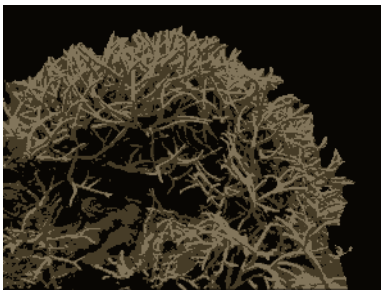

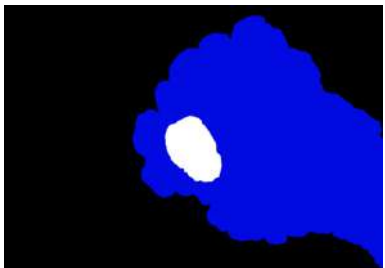
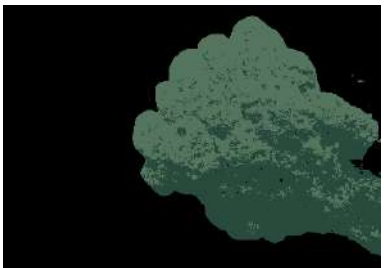

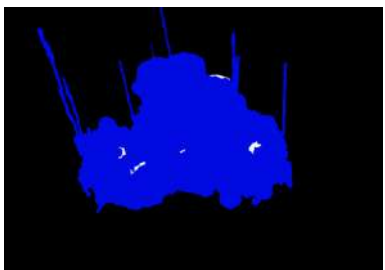
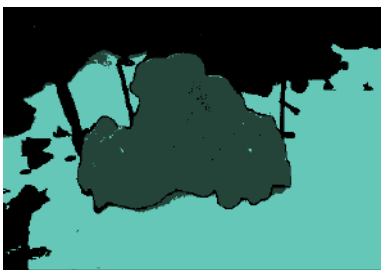

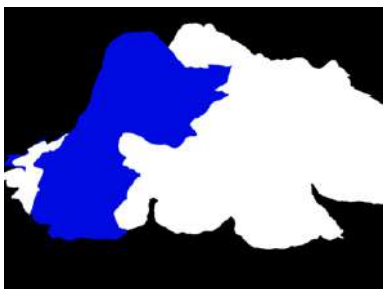
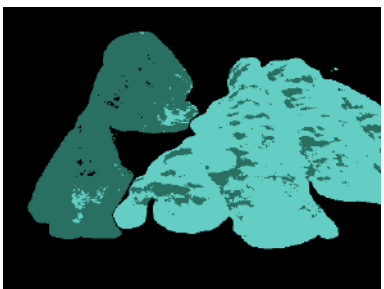



#	Imagen original	Segmentación experto	Segmentación modelo
1			
2			
3			
4			
5			


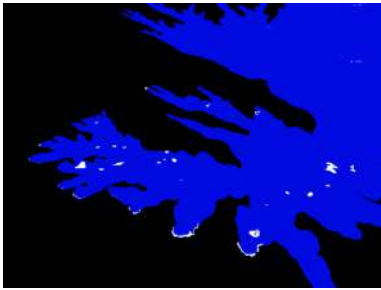


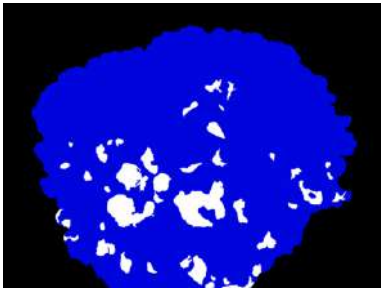
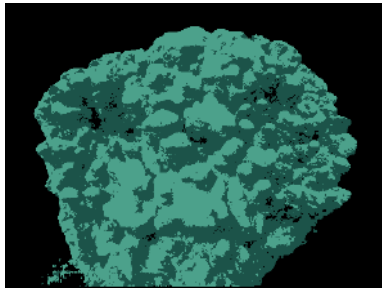


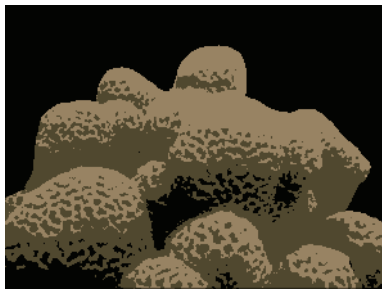

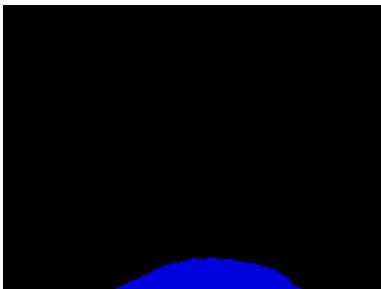


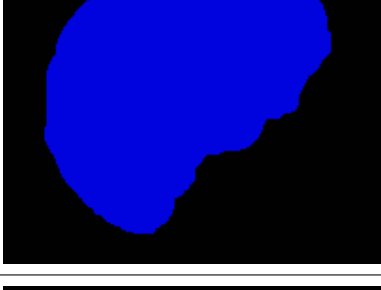
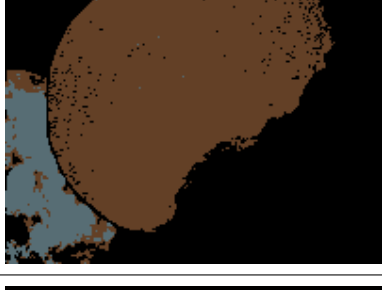
#	Imagen original	Segmentación experto	Segmentación modelo
6			
7			
8			
9			
10			



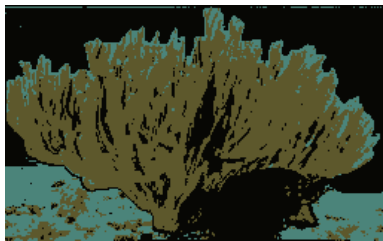


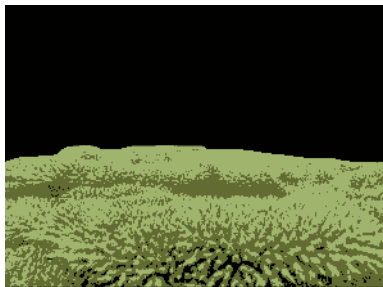
#	Imagen original	Segmentación manual	Segmentación modelo
11			
12			
13			
14			
15			

#	Imagen original	Segmentación manual	Segmentación modelo
16			
17			
18			
19			
20			


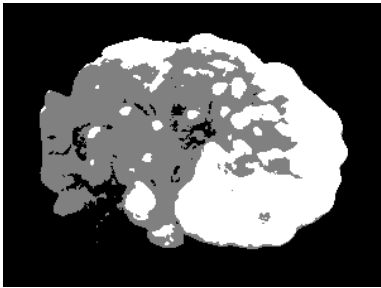
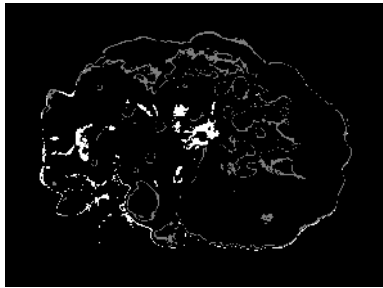
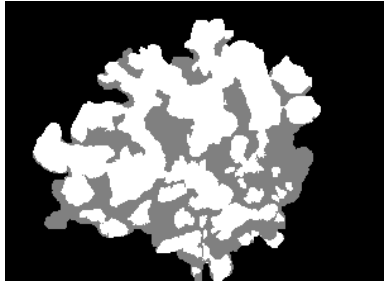
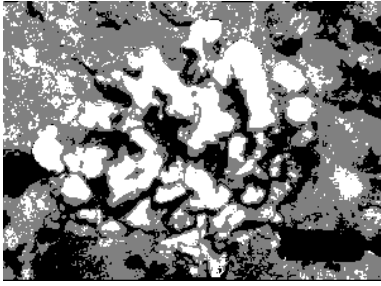
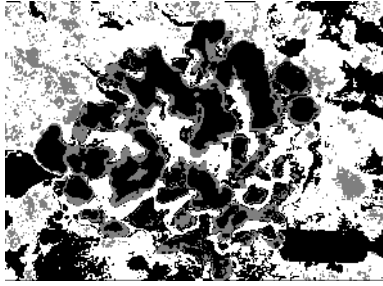
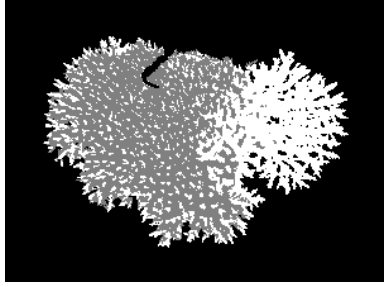
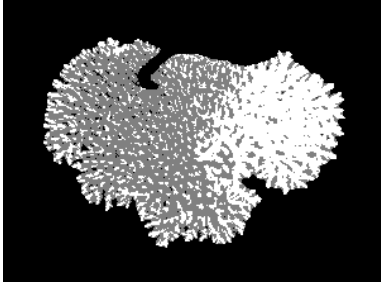
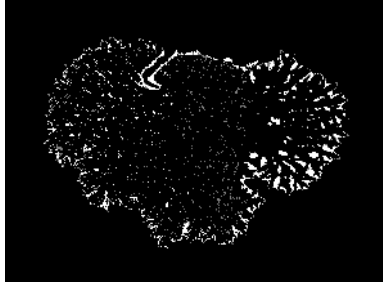

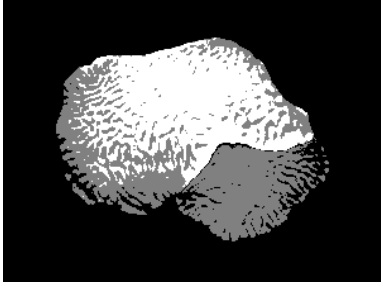

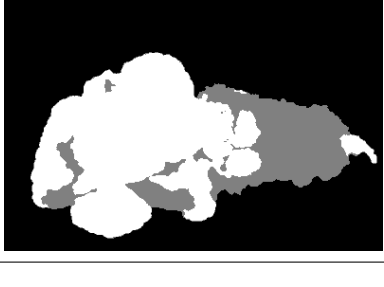
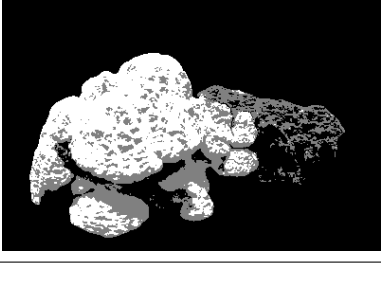
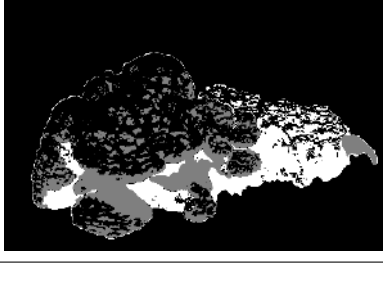
#	Imagen original	Segmentación manual	Segmentación modelo
21			
22			
23			
24			
25			

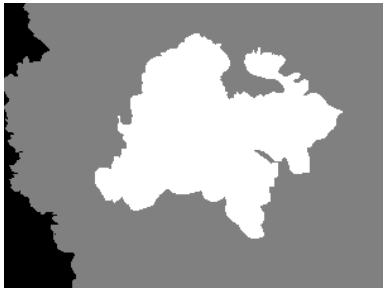
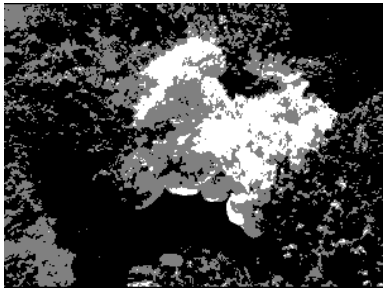


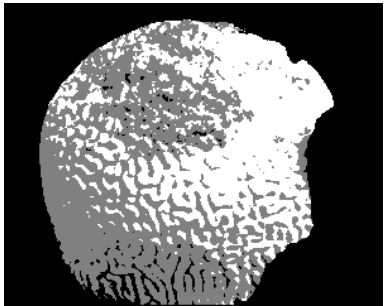
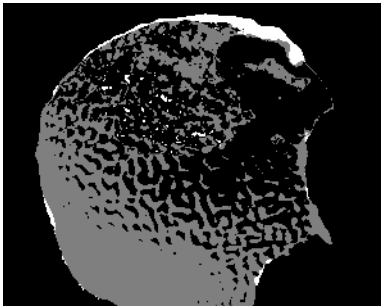




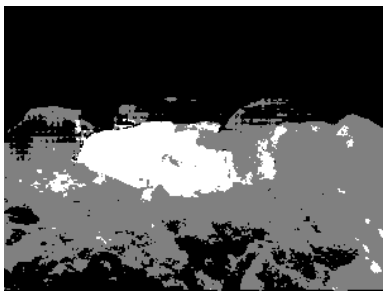


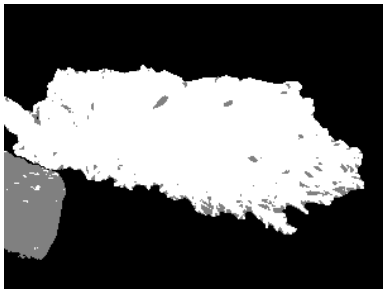

#	Imagen original	Segmentación manual	Segmentación modelo
26			
27			
28			
29			
30			

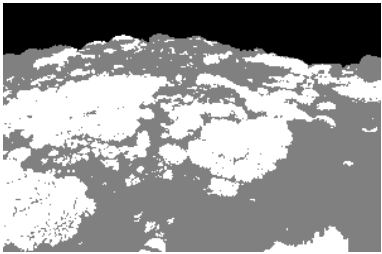
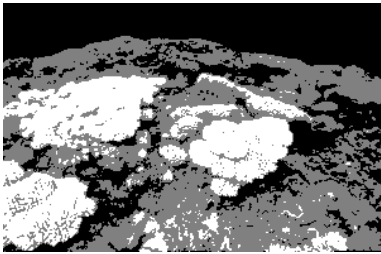
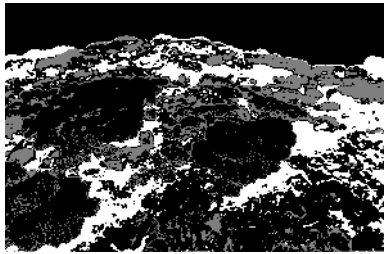
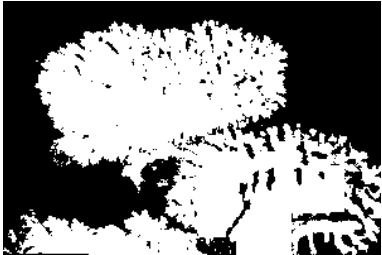
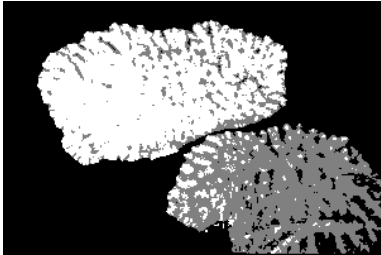
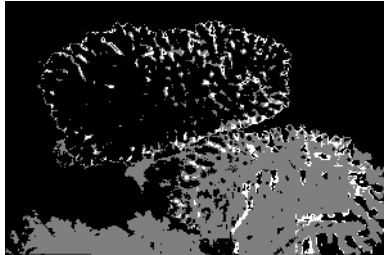
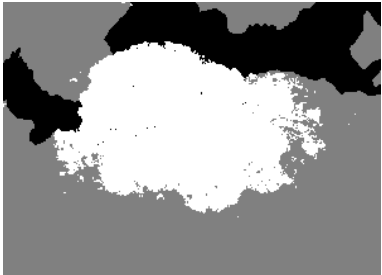
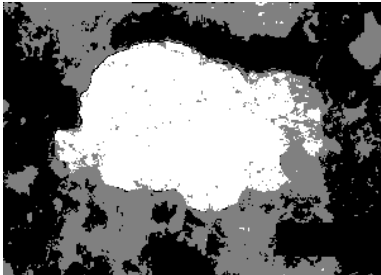
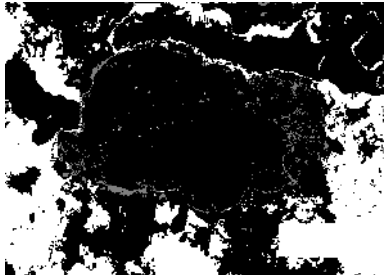



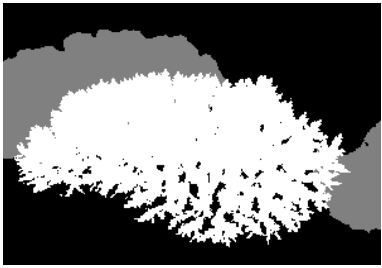
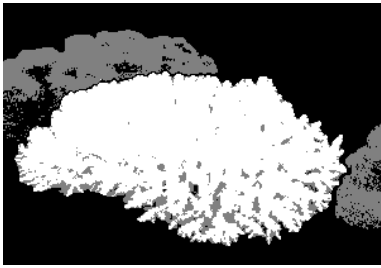

#	Imagen original	Segmentación manual	Segmentación modelo
31			
32			
33			
34			
35			

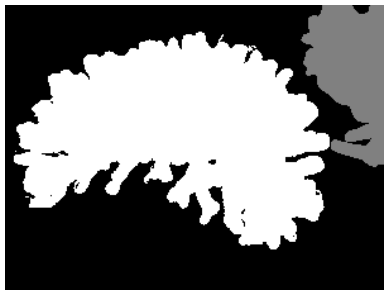
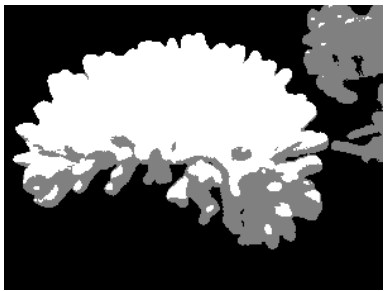

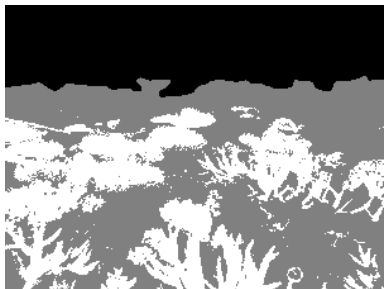
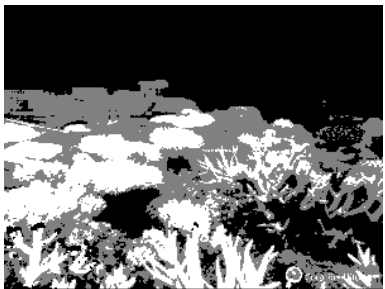
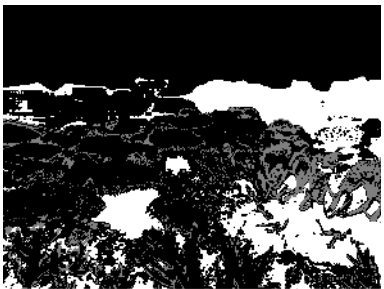

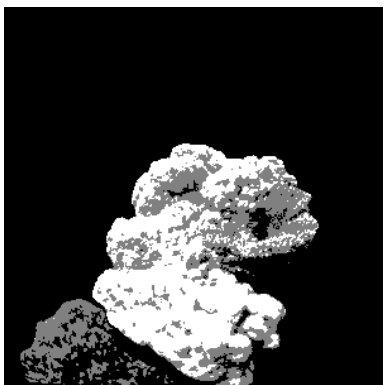

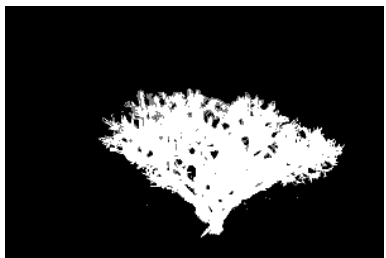


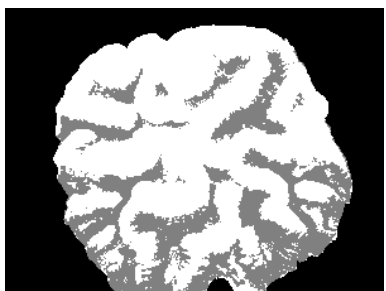

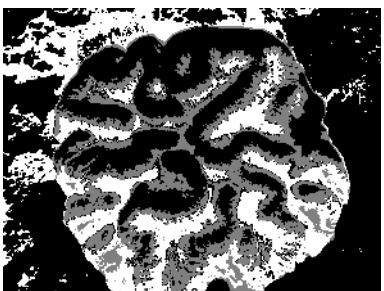
#	Imagen original	Segmentación manual	Segmentación modelo
36			
37			

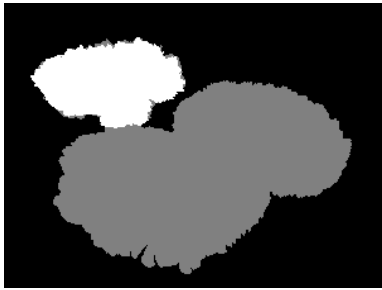
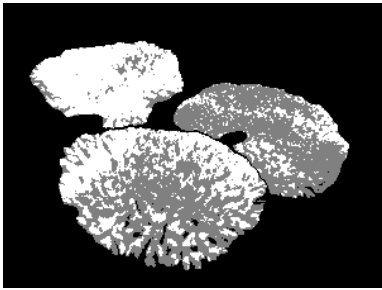
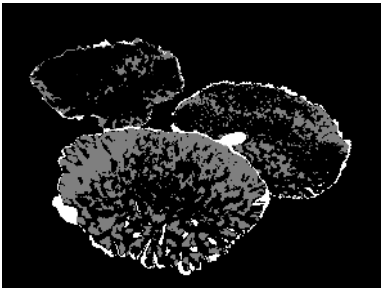
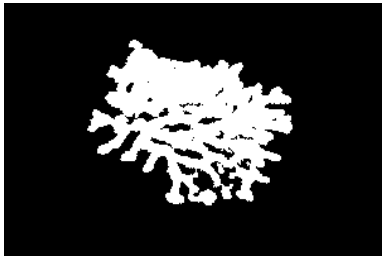
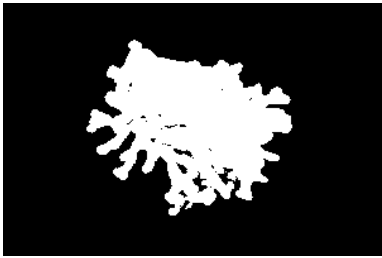
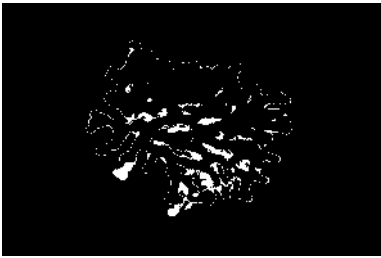
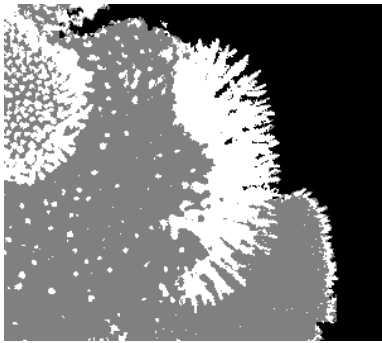
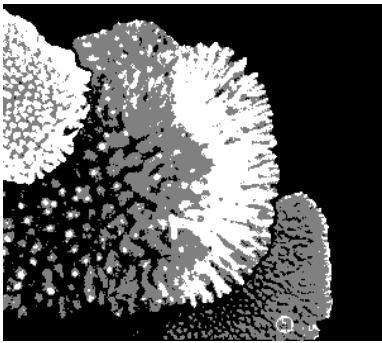
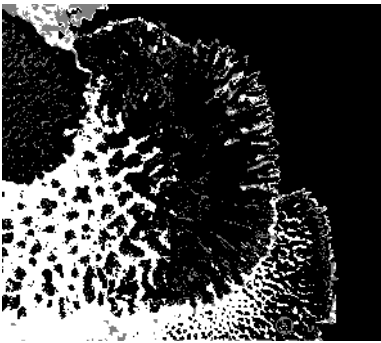



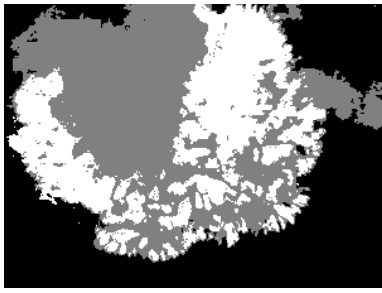
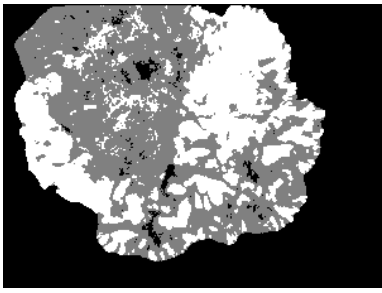
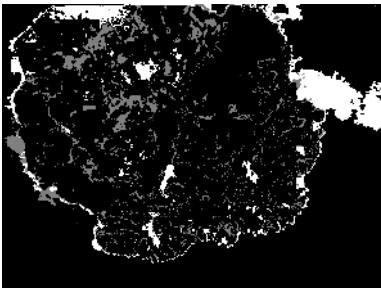
CUADRO 5: Tabla con las imágenes originales, la segmentación manual y la resultante del modelo

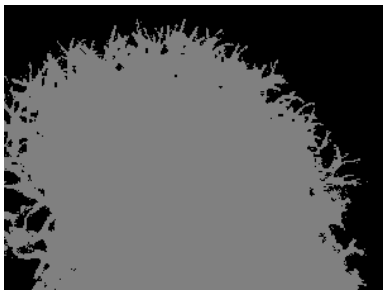
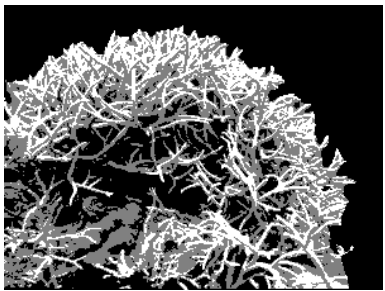
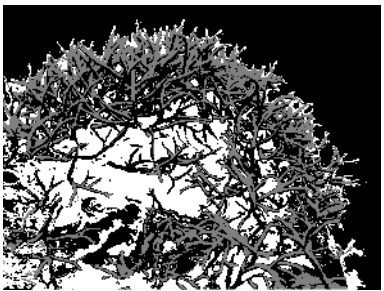

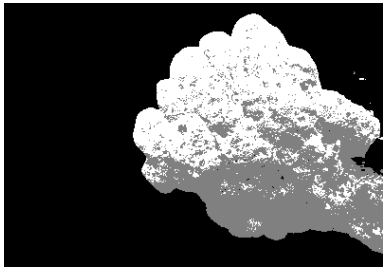
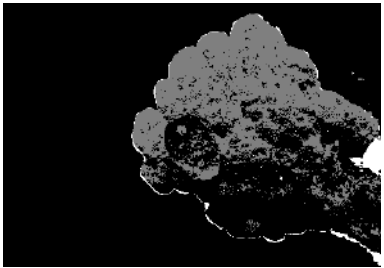

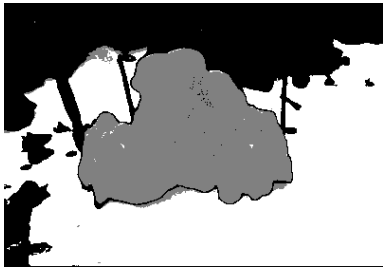
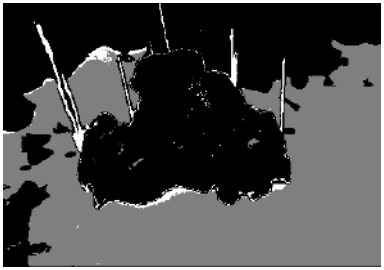
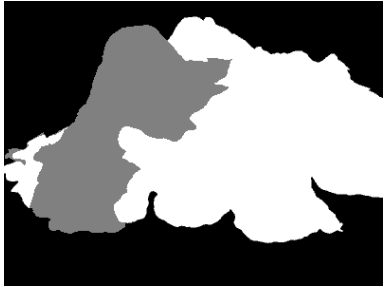
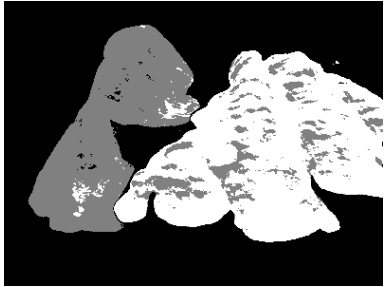


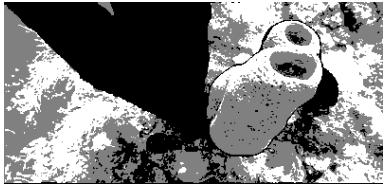
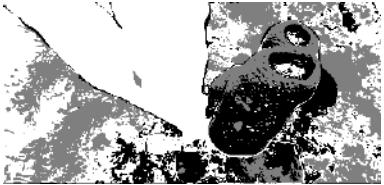
#	Imagen original	Segmentación experto	Diferencias
1			
2			
3			
4			
5			


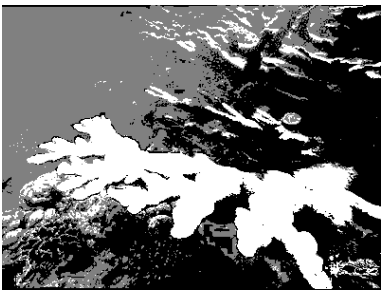
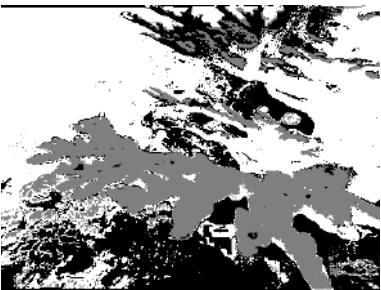
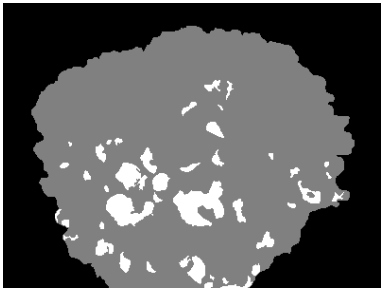
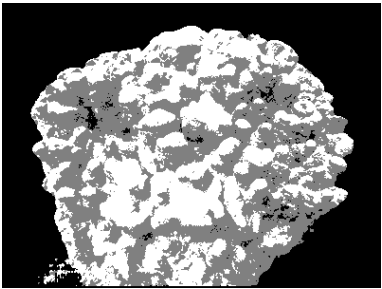
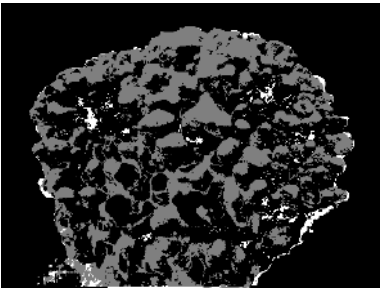

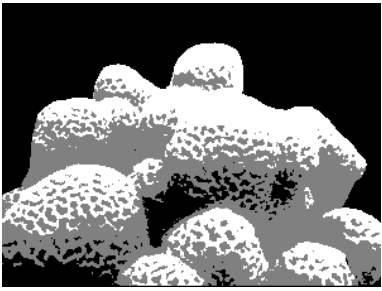


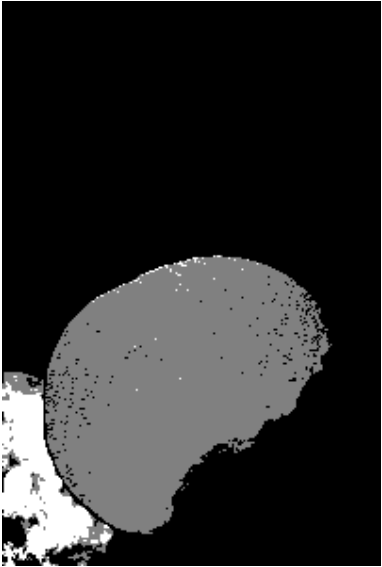
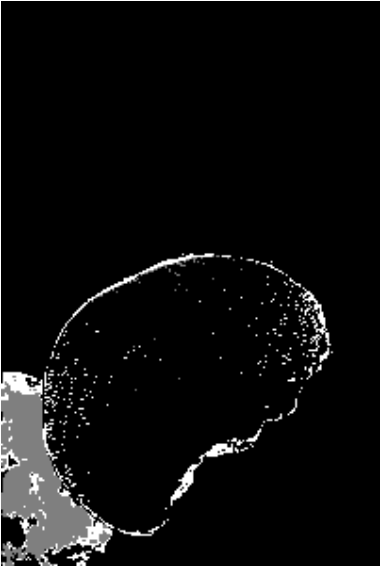

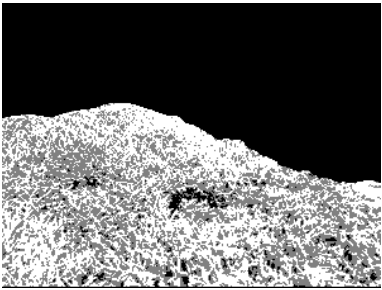
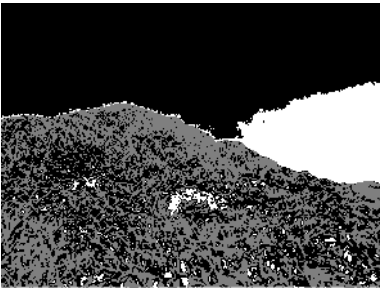
#	Imagen original	Segmentación experto	Diferencias
6			
7			
8			
9			
10			


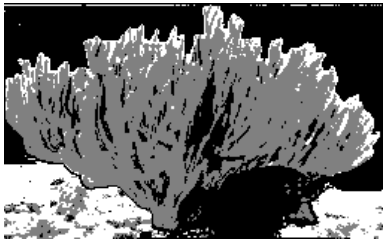
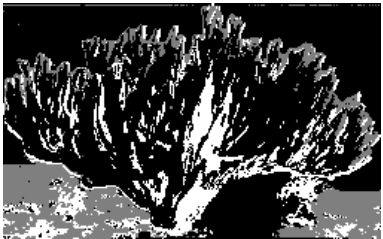

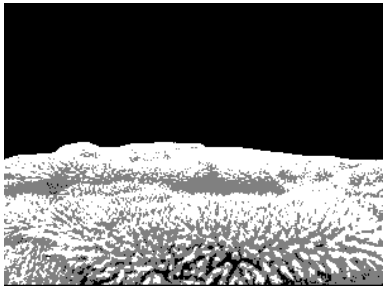

#	Imagen original	Segmentación experto	Diferencias
11			
12			
13			
14			
15			

#	Imagen original	Segmentación experto	Diferencias
16			
17			
18			
19			
20			

#	Imagen original	Segmentación experto	Diferencias
21			
22			
23			
24			
25			

#	Imagen original	Segmentación experto	Diferencias
26			
27			
28			
29			
30			

#	Imagen original	Segmentación experto	Diferencias
31			
32			
33			
34			
35			

#	Imagen original	Segmentación experto	Diferencias
36			
37			

CUADRO 6: Tabla de segmentación hecha por el experto, la resultante del modelo y las diferencias en las imágenes pixel a pixel

Resultados del modelo			
#Imagen	% Blanqueamiento del experto	% Blanqueamiento del modelo	% Error
1	52.02	47.60	9.29
2	61.06	32.26	89.27
3	45.02	45.28	0.56
4	74.76	48.71	53.49
5	68.53	57.06	20.10
6	26.44	27.48	3.77
7	73.33	45.32	61.79
8	13.62	28.95	52.97
9	12.59	19.03	33.86
10	100.00	83.85	19.26
11	40.30	35.79	12.61
12	100.00	53.82	85.82
13	33.94	45.56	25.50
14	100.00	73.32	36.38
15	69.56	69.55	0.01
16	82.42	62.49	31.90
17	37.44	42.44	11.78
18	37.09	54.53	31.98
19	87.79	100.00	12.21
20	68.67	43.37	58.35
21	20.38	48.61	58.06
22	100.00	100.00	0.00
23	30.85	42.34	27.13
24	100.00	14.90	571.04
25	39.93	43.37	7.94
26	100.00	36.71	172.38
27	6.47	49.86	87.03
28	0.71	66.89	98.94
29	66.38	57.91	14.63
30	0.91	41.67	97.81
31	0.82	37.99	97.84
32	9.72	47.98	79.75
33	100.00	43.92	127.70
34	100.00	13.76	626.74
35	100.00	53.04	88.53
36	100.00	27.40	264.90
37	100.00	56.87	75.83

CUADRO 7: Tabla de resultados y error del modelo

Bibliografía

- [SAM] Segment Anything foundation model image segmentation. Accessed: 2024-04-03.
- [nvi] What is a transformer model? Accessed: 2024-04-10.
- [3] (2023). Healthy and bleached corals image classification. Accessed: 2024-01-10.
- [4] (2023). Object masks from prompts with sam. Accessed: 2024-04-10.
- [5] A, R. Q. J. and I, C. M. M. (2011). Redes neuronales artificiales para el procesamiento de imágenes, una revisión de la última década. 9.
- [6] Ajit, A., Acharya, K., and Samanta, A. (2020). A review of convolutional neural networks. *International Conference on Emerging Trends in Information Technology and Engineering, ic-ETITE 2020*.
- [7] Albawi, S., Mohammed, T. A., and Al-Zawi, S. (2017). Understanding of a convolutional neural network: Akdeniz university, antalya, turkey.
- [8] Arthur, D. and Vassilvitskii, S. (2007). K-means++: The advantages of careful seeding. volume 8, pages 1027–1035.
- [9] Barnes, B. B., Hallock, P., Hu, C., Muller-Karger, F., Palandro, D., Walter, C., and Zepp, R. (2015). Prediction of coral bleaching in the florida keys using remotely sensed data. *Coral Reefs*, 34:491–503.
- [10] Calvo, J. (2020). Crear red neuronal desde las matemáticas. european valley. [Internet; recuperado el 3-febrero-2023].
- [11] Carrión, C. B. (2020). Redes convolucionales.
- [12] Chow, M., Tsang, R. H., Lam, E. K., and Ang Jr, P. (2016). Quantifying the degree of coral bleaching using digital photographic technique. *Journal of Experimental Marine Biology and Ecology*, 479:60–68.
- [13] de Jesús Ramírez Alonso, G. M. and Murguía, M. I. C. (2005). Clasificación de defectos en madera utilizando redes neurales artificiales wood defects classification using artificial neural network. 9:17–027.
- [14] Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., Liu, T., Wang, X., Wang, G., Cai, J., and Chen, T. (2018). Recent advances in convolutional neural networks. *Pattern Recognition*, 77:354–377.
- [15] Gómez-Ríos, A., Tabik, S., Luengo, J., Shihavuddin, A. S. M., and Herrera, F. (2019). Coral species identification with texture or structure images using a two-level classifier based on convolutional neural networks . 184:104891.
- [16] Hedley, J. D., Roelfsema, C. M., Chollett, I., Harborne, A. R., Heron, S. F., Weeks, S. J., Skirving, W. J., Strong, A. E., Eakin, C. M., Christensen, T. R., Ticzon, V., Bejarano, S., and Mumby, P. J. (2016). Remote sensing of coral reefs for monitoring and management: A review. *Remote Sensing*, 8.
- [17] Hernández, D. J. (2021). Principales amenazas de los arrecifes de coral.
- [18] Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., and Adam, H. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications.
- [19] Hughes, T. P., Baird, A. H., Bellwood, D. R., Card, M., Connolly, S. R., Folke, C., Grosberg, R., Hoegh-Guldberg, O., Jackson, J. B. C., Kleypas, J., Lough, J. M., Marshall, P., Nyström, M., Palumbi, S. R., Pandolfi, J. M., Rosen, B., and Roughgarden, J. (2003). Climate change, human impacts, and the resilience of coral reefs.
- [20] Izaurieta, F. and Saavedra, C. (2000). Redes neuronales artificiales.

- [21] Jiang, J., Trundle, P., and Ren, J. (2010). Medical image analysis with artificial neural networks. *Computerized Medical Imaging and Graphics*, 34:617–631.
- [22] Josephitis, E., Wilson, S., Moore, J. A., and Field, S. (2012). Comparison of three digital image analysis techniques for assessment of coral cover and bleaching. *Conservation Science Western Australia*, 8(2):251–257.
- [23] Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A. C., Lo, W.-Y., Dollár, P., and Girshick, R. (2023). Segment anything.
- [24] Kondo, C. and Kondo, T. and Ueno, J. (2009). Three-dimensional medical image analysis of the heart by the revised gmdh-type neural network self-selecting optimum neural network architecture. 14:123–128.
- [25] Larranaga, P., Inza, I., and Moujahid, A. (1997). Tema 8. redes neuronales.
- [26] Lesser, M. P. (2011). *Coral bleaching: Causes and mechanisms*. Springer Netherlands.
- [27] Liu, S., Zeng, Z., Ren, T., Li, F., Zhang, H., Yang, J., Li, C., Yang, J., Su, H., Zhu, J., et al. (2023). Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv preprint arXiv:2303.05499*.
- [28] MacKay, D. (2003). An example inference task: Clustering. *Information theory, inference and learning algorithms*, 20:284–292.
- [29] MacQueen, J. et al. (1967). Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, pages 281–297. Oakland, CA, USA.
- [30] Matich, D. J. (2001). Redes neuronales: Conceptos básicos y aplicaciones.
- [31] Minderer, M., Gritsenko, A., Stone, A., Neumann, M., Weissenborn, D., Dosovitskiy, A., Mahendran, A., Arnab, A., Dehghani, M., Shen, Z., Wang, X., Zhai, X., Kipf, T., and Houlsby, N. (2022). Simple open-vocabulary object detection with vision transformers.
- [32] Plaisance, L., Caley, M. J., Brainard, R. E., and Knowlton, N. (2011). The diversity of coral reefs: What are we missing? *PLoS ONE*, 6.
- [33] Portiansky, E. L. (2013). Análisis multidimensional de imágenes digitales. 1.
- [34] Ramos, J. M. S. (2022). Introducción a las redes neuronales artificiales.
- [35] Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. (2016). You only look once: Unified, real-time object detection.
- [36] Ruiz Fernández, L. (2020). Aplicación de filtros morfológicos en imágenes.
- [37] Sully, S., Burkepile, D. E., Donovan, M. K., Hodgson, G., and van Woesik, R. (2019). A global analysis of coral bleaching over the past two decades. *Nature Communications*, 10.
- [38] Suzuki, K. (2011). Artificial neural networks : methodological advances and biomedical applications. page 362.
- [39] Suzuki, K., Horiba, I., Ikegaya, K., and Nanki, M. (1995). Recognition of coronary arterial stenosis using neural network on dsa system. 26:66–74.
- [40] Tan, M., Pang, R., and Le, Q. V. (2020). Efficientdet: Scalable and efficient object detection.
- [41] Wang, X., Zhang, X., Cao, Y., Wang, W., Shen, C., and Huang, T. (2023). Seggpt: Segmenting everything in context.
- [42] Zhang, Z., Liu, L., Kim, J., and Sangaiah, A. (2023). A novel deep multi-instance convolutional neural network for disaster classification from high-resolution remote sensing images. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 17:1–17.
- [43] Zhong, J., Li, M., Qin, J., Cui, Y., Yang, K., and Zhang, H. (2022). Real-time marine animal detection using yolo-based deep learning networks in the coral reef ecosystem. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLVI-3/W1-2022:301–306.
- [44] Zou, X., Yang, J., Zhang, H., Li, F., Li, L., Wang, J., Wang, L., Gao, J., and Lee, Y. J. (2023). Segment everything everywhere all at once.