

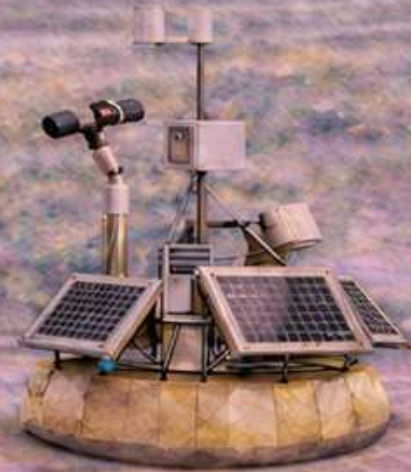
UNIVERSIDAD
DE ANTIOQUIA
1803



Seminario GeoLimna–GEPAR

— Volumen 2 —

Investigaciones 2023



Coordinación

Juan Pablo Serna | David Fernández
Coordinadores de GeoLimna y GEPAR

Colección Seminario Geolimna–GEPAR

© 2026 Universidad de Antioquia

© 2026 Facultad de Ingeniería, Universidad de Antioquia

© 2026 Grupos de investigación Geolimna y GEPAR

ISSN Digital: *(por asignar)*

Queda hecho el depósito que ordena la Ley
(Ley 44 de 1993 / Decreto 460 de 1995)

Dirección editorial: Juan Pablo Serna y David Fernández

Dirección de colección: *Juan Pablo Serna y David Fernández*

Diseño de cubierta y diagramación: *Juan Pablo Serna y David Fernández*

Universidad de Antioquia – Facultad de Ingeniería

Calle 67 # 53–108

Medellín, Colombia

Código Postal: 050010

Correo electrónico: *jpablo.serna@udea.edu.co; david.fernandez@udea.edu.co*

Sitio web: *<https://seminariogg.github.io/Seminario-Geolimna-GEPAR/>*

No está permitida la reproducción total o parcial de este libro ni su tratamiento informático, ni la transmisión de ninguna forma o por cualquier medio ya sea electrónico, mecánico, por fotocopia, por registro u otros métodos, sin el permiso previo y por escrito de los titulares de derechos.

Contenido

Prefacio	IV
1 GeoVAM: plataforma web de integración y visualización de variables ambientales en tiempo real	1
2 Clasificación de esquejes de crisantemo mediante visión por computador y aprendizaje automático	16
3 Monitoreo multitemporal de coberturas del suelo en áreas de minería de oro aluvial mediante teledetección y minería de patrones secuenciales	30
4 Laboratorio remoto para enseñanza y experimentación en redes de datos mediante control web de dispositivos	43
5 Estimación objetiva de la frecuencia respiratoria en larvas de megalóptero mediante procesamiento digital de imágenes	55
6 Generación sintética y realista de matrículas vehiculares para entrenamiento de algoritmos de reconocimiento de patrones	68
7 Gestión y despliegue parametrizable de información de grupos de investigación mediante una plataforma web responsiva	81
8 Clasificación asistida de corales en imágenes submarinas mediante visión artificial	94
9 Caracterización multitemporal de coberturas flotantes en embalses mediante procesamiento de imágenes satelitales multiespectrales	107
10 Identificación asistida de tiburones mediante análisis de aletas dorsales y descriptores de Fourier	124
11 Comparación morfométrica multitemporal de un humedal costero mediante análisis batimétrico y modelación espacial	140
12 Caracterización ecohidrobiológica de un humedal costero y registro de una planta carnívora acuática en El Eneal, San Onofre	154

Comité Académico

Facultad de Ingeniería - Universidad de Antioquia

Coordinadores

Juan Pablo Serna López

Profesor asociado, Departamento de Ingeniería Ambiental y Sanitaria.

`jpablo.serna@udea.edu.co`

David Fernández Mc Cann

Profesor titular, Departamento de Ingeniería Electrónica y Telecomunicaciones.

`david.fernandez@udea.edu.co`

Integrantes

Néstor Jaime Aguirre Ramírez

Profesor titular, Departamento de Ingeniería Ambiental y Sanitaria.

`nestor.aguirre@udea.edu.co`

Fabio de Jesús Vélez Macías

Profesor titular, Departamento de Ingeniería Ambiental y Sanitaria. `fabio.velez@udea.edu.co`

Elkin Libardo Ríos Ortiz

Profesor titular, Departamento de Ingeniería Industrial. `elkin.rios@udea.edu.co`

Juan Pablo Urrea Duque

Profesor asociado, Departamento de Ingeniería Electrónica y Telecomunicaciones.

`juan.urrea@udea.edu.co`

Yarin Tatiana Puerta Quintana

Profesora asistente, Departamento de Ingeniería Ambiental y Sanitaria.

`tatiana.puerta@udea.edu.co`

Prefacio

Este volumen del *Seminario Geolimna–GEPAR* da continuidad al trabajo académico, técnico e institucional que ambos grupos han venido consolidando en los últimos años. Si la entrega anterior permitió organizar los desarrollos comprendidos entre 2019 y 2021, el presente tomo reúne una etapa posterior, correspondiente a 2023, en la que varios proyectos muestran mayor definición metodológica, mejores esquemas de validación y una proyección más clara hacia el uso, la transferencia y la continuidad.

Durante este periodo, la articulación entre Geolimna y GEPAR siguió expresándose en desarrollos orientados a problemas reales de observación, medición, automatización y análisis. Más que una coincidencia entre áreas, lo que aquí se documenta es una forma de trabajo en la que preguntas de carácter ambiental, biológico o productivo se traducen en métodos, dispositivos, algoritmos, software y plataformas de apoyo técnicamente verificables. En esa dinámica, la ingeniería no aparece separada del contexto, sino vinculada de manera directa con las necesidades que motivan cada desarrollo.

Este libro mantiene la orientación editorial de la colección: reunir en un solo volumen una memoria técnica organizada de los trabajos presentados en el seminario. Los capítulos han sido escritos como reportes de investigación y desarrollo, con el propósito de dejar registro del problema abordado, del enfoque metodológico adoptado, de la implementación realizada, de los resultados obtenidos y de sus alcances. Esta forma de documentación busca facilitar la continuidad de los procesos y ofrecer una base útil para trabajos posteriores de ajuste, comparación, escalamiento o transferencia.

No todos los capítulos corresponden a artículos científicos en sentido estricto, ni pretenden sustituir ese tipo de publicación. En varios casos, el valor de este volumen está precisamente en recoger materiales que suelen quedar dispersos entre presentaciones, manuales, tesis, registros de software, informes y documentos internos. Al integrarlos en un formato común, el libro permite conservar una visión más completa del proceso técnico que dio lugar a cada resultado.

En el conjunto del volumen aparece una idea compartida: muchos problemas ambientales requieren hoy sistemas de observación, medición y análisis cada vez más precisos, y esa

exigencia demanda soluciones de ingeniería. A su vez, la ingeniería encuentra aquí un campo de aplicación concreto, donde sensores, instrumentación, procesamiento de imágenes, análisis de datos, automatización y plataformas computacionales se orientan a comprender mejor un fenómeno, apoyar decisiones o mejorar un proceso específico.

Este tomo también deja ver un rasgo importante de maduración. Varios de los desarrollos aquí presentados ya no se reducen a ejercicios exploratorios o pruebas aisladas, sino que forman parte de una capacidad instalada construida de manera progresiva. La existencia de manuales, registros, metodologías de captura, software especializado y articulaciones con docencia, investigación y extensión muestra una producción más acumulativa y reutilizable, y esa continuidad justifica plenamente la consolidación de este nuevo volumen.

En el ámbito interno, el libro busca servir como memoria organizada del seminario y como punto de partida para nuevas cohortes de estudiantes, docentes e investigadores. Hacia el exterior, ofrece una muestra del tipo de resultados que pueden surgir cuando la colaboración entre grupos se sostiene en problemas bien definidos, trabajo técnico riguroso y una clara orientación hacia aplicaciones verificables. En ese sentido, este volumen aspira a dejar constancia no solo de unos productos, sino también de una manera de construirlos.

Grupos Geolimna y GEPAR
Universidad de Antioquia
Volumen 2, Investigaciones 2023

Capítulo 1

GeoVAM: plataforma web de integración y visualización de variables ambientales en tiempo real

Autores

Cristian Alexis Quintana Valencia¹; John Fernando Ramírez Gómez²; Julián Muñoz³; Sara Ospina⁴; David Fernández Mc Cann⁵; Fabio de Jesús Vélez Macías⁶; Néstor Jaime Aguirre Ramírez⁷.

Grupos GeoLimna y GEPAR, Facultad de Ingeniería, Universidad de Antioquia

Resumen

GeoVAM es la capa web integradora del ecosistema SIMEVAM, concebida para publicar, organizar y hacer interpretable la información proveniente de estaciones ambientales distribuidas y de los sistemas instrumentales asociados al monitoreo de la ciénaga de Ayapel

¹Estudiante de Ingeniería Electrónica y Telecomunicaciones.

²Estudiante de Ingeniería Electrónica y Telecomunicaciones.

³Estudiante de Ingeniería Electrónica y Telecomunicaciones.

⁴Estudiante de Ingeniería Ambiental y Sanitaria.

⁵Profesor titular, Departamento de Ingeniería Electrónica y Telecomunicaciones.

⁶Profesor titular, Departamento de Ingeniería Ambiental y Sanitaria.

⁷Profesor titular, Departamento de Ingeniería Ambiental y Sanitaria.

^{1,2,3,4,5,6,7}Universidad de Antioquia. Grupos de investigación GEPAR y GeoLimna.

y otros contextos de observación. Más que una interfaz de consulta, GeoVAM articula adquisición, transmisión, almacenamiento y visualización de datos ambientales en tiempo real. La documentación disponible muestra que la plataforma despliega las estaciones en un mapa y en una tabla, permite explorar series temporales multivariadas con filtros y normalización, acceder a páginas descriptivas por estación y consultar videos de cobertura nubosa asociados a sistemas particulares. En este capítulo se presenta la arquitectura funcional de GeoVAM y se sitúan dentro de ella distintos aportes e iniciativas vinculados al proyecto SIMEVAM, entre ellos los desarrollos asociados a Cristian Alexis Quintana Valencia, John Fernando Ramírez Gómez, Sara Ospina y Julián Muñoz. Se propone además una formulación matemática mínima para representar la lógica de integración de datos heterogéneos en una misma plataforma de observación ambiental.

Palabras clave: plataforma web; variables ambientales; series de tiempo; integración de datos; SIMEVAM; IoT ambiental; GeoVAM; visualización en tiempo real.

1.1. Introducción

En muchos proyectos de monitoreo ambiental, la dificultad principal no termina cuando un sensor mide correctamente. En realidad, apenas comienza allí. Un sistema de observación madura cuando logra convertir señales aisladas, obtenidas por dispositivos heterogéneos y situados en condiciones de campo diversas, en información accesible, interpretable y trazable para investigadores, comunidades y tomadores de decisión. GeoVAM surge precisamente en ese punto: no como un sensor ni como una estación, sino como la plataforma que permite que un conjunto de desarrollos instrumentales y computacionales opere como red.

La documentación del proyecto describe a GeoVAM como un software que permite observar gráficamente datos e información de diferentes variables ambientales, en tiempo real, recopilados por estaciones situadas en distintas regiones del país. Estas estaciones se muestran en un mapa y en una tabla, desde donde el usuario puede acceder a gráficos temporales, filtros por periodo, opciones de normalización, páginas de detalle por estación y repositorios de video asociados a cobertura nubosa [1, 2]. Esta descripción muestra que GeoVAM no fue concebido para una única variable ni para un único dispositivo, sino para una arquitectura de observación distribuida y extensible.

Dentro del marco SIMEVAM, esta plataforma adquiere un papel aún más relevante. El acta de inicio y propiedad intelectual del proyecto establece que el sistema de registro y seguimiento de variables ambientales en la ciénaga de Ayapel busca medir variables físicas, químicas y biológicas mediante instrumentación instalada *in situ*, así como construir

1 GeoVAM: plataforma web de integración y visualización de variables ambientales en tiempo real

modelos de relación con el pulso de inundación y apoyar pronósticos ambientales [3]. En esa lógica, la medición de campo, el sensado multivariable, la comunicación inalámbrica, el almacenamiento en la nube y la visualización no son actividades independientes, sino partes de una misma cadena técnica e institucional. GeoVAM corresponde al tramo en el cual dicha cadena se vuelve pública, navegable y útil.

GeoVAM puede leerse, además, como una síntesis funcional de varios aportes realizados por estudiantes y desarrollos asociados al universo SIMEVAM. Algunos de esos aportes desembocaron en herramientas específicas, como SIREMEVAM, SUCA, CorrentómetroBT o sistemas de videomonitorio, mientras que otros corresponden a trabajos de integración, transmisión, despliegue o diseño de interfaz. La plataforma adquiere sentido precisamente al reunir y ordenar esas trayectorias dentro de una misma estructura operativa.



Figura 1.1: Pantalla principal de GeoVAM (geovam.udea.edu.co). La interfaz presenta las estaciones sobre un mapa, una tabla de acceso y enlaces hacia gráficos, páginas de detalle y material multimedia.

1.2. Problema y justificación del desarrollo

El problema técnico al que responde GeoVAM puede formularse así: dado un conjunto de estaciones ambientales y subsistemas de medición distribuidos, cada uno con variables, frecuencias de muestreo y rutas de transmisión potencialmente diferentes, se requiere una plataforma capaz de unificar la publicación, consulta y exploración de dicha información en tiempo real, preservando contexto de estación, trazabilidad temporal y acceso diferenciado a recursos asociados.

Formalmente, si se considera un conjunto de estaciones

$$\mathcal{E} = \{E_1, E_2, \dots, E_M\},$$

y para cada estación E_m un conjunto de variables ambientales medidas

$$\mathcal{V}_m = \{v_{m,1}, v_{m,2}, \dots, v_{m,K_m}\},$$

entonces GeoVAM debe construir una representación funcional que permita consultar, para una estación E_m y una variable $v_{m,k}$, la serie temporal

$$x_{m,k}(t) = \{x_{m,k}(t_1), x_{m,k}(t_2), \dots, x_{m,k}(t_n)\},$$

donde cada muestra queda asociada a una fecha, una unidad y un contexto instrumental específico.

La dificultad no es solo almacenar el conjunto de muestras. También es necesario hacer convivir en una misma plataforma variables heterogéneas, por ejemplo temperatura, humedad, pH, oxígeno disuelto, conductividad, profundidad, velocidad de flujo o cualquier otra magnitud que los desarrollos SIMEVAM hayan logrado transmitir. Si se define para cada estación un vector multivariable de observación

$$\mathbf{x}_m(t) = (x_{m,1}(t), x_{m,2}(t), \dots, x_{m,K_m}(t)),$$

la plataforma debe permitir al usuario seleccionar componentes, escalarlas, filtrarlas y, cuando sea útil, normalizarlas para comparación visual. Una normalización simple puede expresarse como

$$\tilde{x}_{m,k}(t) = \frac{x_{m,k}(t) - \mu_{m,k}}{\sigma_{m,k}},$$

donde $\mu_{m,k}$ y $\sigma_{m,k}$ representan respectivamente la media y la desviación estándar de la variable $v_{m,k}$ en el intervalo observado. Esta expresión no pretende describir todo el procesamiento interno del software, pero sí formaliza una de las operaciones explícitamente

mencionadas en el manual de usuario [1]: la posibilidad de normalizar series para facilitar su lectura comparada.

Desde el punto de vista práctico, GeoVAM resuelve además un problema de dispersión. Sin una plataforma de este tipo, cada estación tendería a quedar encapsulada en su propio flujo de datos, en su propio formato y en su propia interfaz. La consecuencia sería una colección de soluciones parciales, no una red de monitoreo. GeoVAM corrige esa fragmentación al convertir múltiples trayectorias técnicas en un único sitio de acceso.

1.3. Contexto institucional: SIMEVAM como marco de integración

El acta de registro y propiedad intelectual del proyecto SIMEVAM deja claro que el propósito central del sistema fue medir variables ambientales en tiempo real en el sistema cenagoso de Ayapel, relacionarlas con el pulso de inundación y construir bases para pronóstico y modelación [3]. También se explicita un compromiso de circulación del conocimiento con la comunidad, incluyendo talleres PRACCIS y un cierre en el cual la comunidad pudiera continuar con las mediciones y el uso del sistema en Ayapel. Esta orientación permite entender que GeoVAM no es un desarrollo aislado de ingeniería de software; es una pieza diseñada para sostener continuidad de medición, acceso y apropiación.

En una ampliación posterior del mismo ecosistema, el documento de SIMEVAM II plantea explícitamente complementar SIMEVAM I con nuevos sensores de calidad del aire y del agua, integrarlos a la infraestructura existente y entregar los datos obtenidos a GeoVAM [4]. En otras palabras, el propio proyecto reconoce a la plataforma como destino natural de integración. El objetivo ya no es solamente medir, sino integrar lo medido en un entorno disponible en la nube y visible desde navegadores y dispositivos móviles.

Esta decisión muestra que la plataforma web no debe verse como un desarrollo accesorio, sino como parte de la arquitectura general de observación ambiental. En términos sistémicos, si se representa el flujo completo como una composición de bloques,

$$\mathcal{S} = \mathcal{A} \circ \mathcal{T} \circ \mathcal{B} \circ \mathcal{W},$$

donde \mathcal{A} representa adquisición, \mathcal{T} transmisión, \mathcal{B} persistencia en base de datos y \mathcal{W} visualización web, GeoVAM se ubica en el bloque \mathcal{W} , pero solo cobra sentido como resultado de la cadena completa. De igual manera, su diseño condiciona la forma en que los demás bloques organizan y publican los datos.

1.4. Arquitectura funcional de GeoVAM

A partir de los manuales técnico y de usuario, la arquitectura funcional de GeoVAM puede entenderse como una secuencia de cuatro niveles que trabajan de manera acoplada. El primero corresponde al nivel de adquisición y procedencia de datos: estaciones y sistemas instrumentales que publican variables ambientales. El segundo corresponde al acceso y sincronización de datos desde servicios en línea, hojas de cálculo y base de datos. El tercero organiza la lógica del servidor y los controladores. El cuarto materializa la experiencia de usuario mediante vistas web, tablas, mapas, gráficos y páginas de detalle [1, 2].

Esta arquitectura puede expresarse conceptualmente así. Sea E_m una estación y sea D_m su conjunto de datos publicados. La plataforma implementa una función de recuperación

$$R(E_m) = D_m,$$

una función de transformación visual

$$\Phi(D_m) = \{G_m, P_m, V_m\},$$

donde G_m representa gráficos temporales, P_m los metadatos o página descriptiva de la estación y V_m los recursos de video asociados, y una función de despliegue final

$$\Psi(E_m) = (M_m, T_m, G_m, P_m, V_m),$$

donde M_m es la representación en mapa y T_m la representación tabular de acceso.

GeoVAM no reduce la estación a un solo tipo de salida. Cada estación es simultáneamente un punto geográfico, una entidad tabulada, una fuente de series temporales, una ficha descriptiva y, en algunos casos, un origen de video. Esa multifuncionalidad acerca la plataforma a una infraestructura de observación ambiental más que a un simple tablero de indicadores.

El manual técnico menciona además módulos y controladores específicos, entre ellos `googleController`, `serverController` y `videoController`, junto con rutas asociadas a acceso de datos y manejo de interfaz [2]. Sin entrar a detallar un código que en esta tanda documental no fue entregado completo, sí puede sostenerse que la aplicación fue organizada en módulos diferenciados para acceso a datos, servicios del servidor, videos y vistas de interfaz. Esa modularidad permite ampliar estaciones, variables o servicios sin reescribir toda la plataforma.



Figura 1.2: Vista y organización funcional de GeoVAM según el manual técnico. Se observa la transición entre la capa de datos, la lógica de servidor y la interfaz gráfica accesible desde navegador.

1.5. Funciones principales de la plataforma

La descripción funcional del manual de usuario permite distinguir con claridad el funcionamiento principal de GeoVAM [1]. El usuario primero identifica dónde están las estaciones; luego explora qué variables producen; después analiza su comportamiento temporal; y finalmente consulta contexto ampliado mediante detalles y videos.

La página de inicio cumple el papel de índice espacial y tabular del sistema. Allí las estaciones aparecen en un mapa y también en una tabla, con lo cual se ofrecen dos modos de entrada a la información: uno geográfico y otro estructurado. Esta doble representación es importante porque algunos usuarios piensan el monitoreo desde el territorio, mientras que otros lo hacen desde el nombre del sistema o la lista de estaciones.

El gráfico de datos constituye la segunda gran función. Si para una estación E_m el usuario

selecciona un subconjunto de variables

$$\mathcal{U}_m \subseteq \mathcal{V}_m,$$

GeoVAM construye una representación temporal conjunta

$$G_m(t) = \{x_{m,k}(t) : v_{m,k} \in \mathcal{U}_m\},$$

y permite interacciones sobre el intervalo temporal y sobre la escala de los datos. Esa operación transforma un conjunto de muestras en una lectura comparativa del comportamiento ambiental. Desde una perspectiva de ingeniería, el valor no está solo en graficar, sino en permitir al usuario elegir qué observar y sobre qué horizonte temporal hacerlo.

La página de detalles por estación agrega un tercer nivel. Allí la estación deja de ser solo una serie y recupera su contexto: descripción, coordenadas geográficas, variables medidas y sentido ambiental de cada variable. Esto es especialmente importante en plataformas interdisciplinarias, donde el mismo dato puede tener un significado instrumental para un ingeniero, ecológico para un limnólogo y operativo para una comunidad local.

Finalmente, GeoVAM incorpora acceso a videos de estaciones. Esta función introduce una capa visual no reducida a la numerización de variables. En particular, el manual menciona videos de cobertura nubosa, organizados por fecha y accesibles en nuevas ventanas [1]. Esta capacidad convierte a la plataforma en un repositorio mixto de observación: series de tiempo por un lado, evidencia visual atmosférica por otro.



Figura 1.3: Gráfico temporal de variables en GeoVAM. Se muestra la lógica de selección de estación, exploración por periodo y comparación de variables, incluyendo la opción de normalización cuando está disponible.

1.6. GeoVAM como síntesis de aportes dentro de SIMEVAM

GeoVAM no condensa únicamente una línea de programación web, sino una convergencia de trabajos y prototipos desarrollados alrededor de SIMEVAM. El propio expediente del proyecto muestra entre sus anexos actas e informes asociados a John Fernando Ramírez Gómez, Julián Muñoz, Cristian Quintana y Sara Ospina, además de otros apoyos estudiantiles y de jóvenes investigadoras. Esa evidencia institucional permite leer la plataforma como una construcción colectiva.

En el caso de **Cristian Alexis Quintana Valencia**, la documentación de SIMEVAM II resulta especialmente relevante. Allí se establece como objetivo complementar SIMEVAM I con nuevos sensores para calidad del aire y del agua, integrar esos dispositivos con la infraestructura existente y entregar los datos obtenidos a GeoVAM desde una plataforma en la nube, explícitamente Firebase. También se señala el desarrollo de una aplicación móvil sencilla en Kodular para visualizar el comportamiento de SIMEVAM II en tiempo real. En consecuencia, su trabajo puede entenderse como un aporte a la expansión de la oferta de variables y a la conexión entre instrumentación nueva, nube y plataforma web [4, 5].

En el caso de **John Fernando Ramírez Gómez**, su participación se hace visible en al menos dos direcciones. Por una parte, aparece como responsable estudiantil del proyecto SUCA, una aplicación móvil para medición superficial cautiva que se articula explícitamente con SIMEVAM [6, 7, 8]. Por otra, el material del taller SIMEVAM y el libro anterior del seminario muestran su presencia en el desarrollo de CorrentómetroBT, sistema inalámbrico para estimación de velocidad de flujo de agua [9, 10, 11]. Aunque ni SUCA ni CorrentómetroBT sean equivalentes a GeoVAM, ambos amplían el universo de instrumentos cuyos datos o contextos pueden integrarse y consultarse en una plataforma mayor.

Sara Ospina también debe ser mencionada como aportante al ecosistema, pues el expediente del proyecto la registra entre los desarrollos y anexos asociados a SIMEVAM. Aun cuando en la documentación disponible no se despliega todavía un manual completo de su software o módulo específico, su presencia institucional indica un trabajo efectivamente incorporado al proyecto y, por tanto, al tejido técnico que alimenta la consolidación de GeoVAM.

Julián Muñoz aparece igualmente como uno de los trabajos de grado anexados a la ejecución del proyecto. En el estado actual de la documentación compartida, no se dispone todavía de una memoria técnica completa de su solución individual, pero su inclusión

dentro de los soportes de SIMEVAM justifica incorporarlo aquí como aportante del ecosistema.

Hay además otros desarrollos que conviene situar alrededor de GeoVAM, aunque no todos pertenezcan estrictamente a la capa web. SIREMEVAM, mencionado en la documentación como aplicación móvil funcional y útil cuando no se dispone de computador, representa una vía de acceso alterna al sistema. SICUANU aporta el componente de videos de cobertura nubosa que GeoVAM es capaz de publicar. SUCA aporta lectura, recepción y almacenamiento móvil de variables acuáticas. CorrentómetroBT y SIMEVAM II amplían la oferta de variables y el repertorio instrumental. De este modo, la plataforma puede entenderse como nodo de integración de varias trayectorias de desarrollo.

1.7. Integración de iniciativas SIMEVAM

GeoVAM puede representarse como un nodo central alrededor del cual se articulan distintos desarrollos que alimentan la red SIMEVAM. La plataforma recibe entradas desde al menos cuatro fuentes: estaciones base de SIMEVAM I, ampliación instrumental de SIMEVAM II, subsistemas móviles de captura y visualización, y módulos de video atmosférico o cobertura nubosa.

$$\mathcal{G} = (\mathcal{N}, \mathcal{A}),$$

donde el conjunto de nodos \mathcal{N} contiene

$$\mathcal{N} = \{\text{SIMEVAM I, SIMEVAM II, SUCA, SIREMEVAM, CorrentómetroBT, SICUANU, GeoVAM}\},$$

y el conjunto de arcos \mathcal{A} representa relaciones de publicación, integración, consulta o complementariedad. Así, por ejemplo, un arco

$$(\text{SIMEVAM II, GeoVAM}) \in \mathcal{A}$$

expresa que los nuevos sensores integrados por SIMEVAM II fueron concebidos para entregar datos a GeoVAM. De igual manera,

$$(\text{SICUANU, GeoVAM}) \in \mathcal{A}$$

representa la integración de videos de nubosidad; y

$$(\text{SUCA, GeoVAM}) \in \mathcal{A}$$

1 GeoVAM: plataforma web de integración y visualización de variables ambientales en tiempo real

se interpreta como complementariedad funcional dentro del mismo ecosistema de monitoreo.

La Figura 1.4 resume esta estructura de integración entre sensado, transmisión, nube, visualización web y acceso móvil.

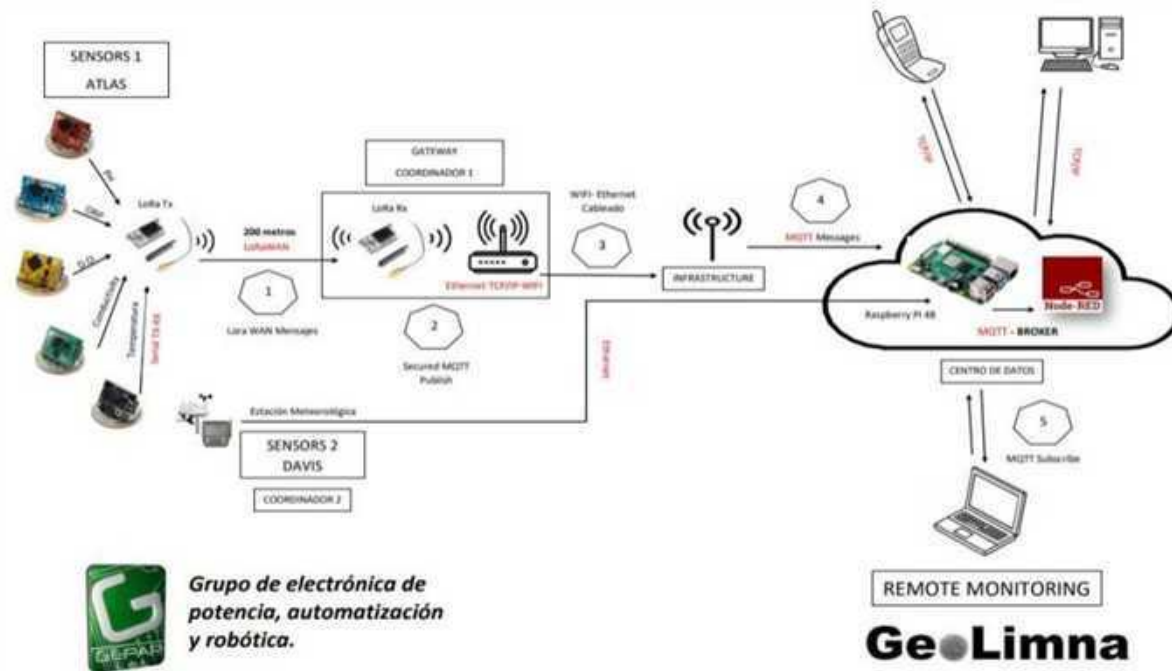


Figura 1.4: Esquema de integración de GeoVAM dentro del ecosistema SIMEVAM. La figura muestra el tránsito desde sensores y estaciones hacia transmisión, nube, visualización web y acceso móvil, con GeoVAM como nodo articulador.

1.8. Resultados funcionales observables

Aunque la documentación disponible no entrega en esta tanda una evaluación cuantitativa exhaustiva del desempeño de GeoVAM en términos de latencia, concurrencia o métricas de usabilidad, sí permite establecer resultados funcionales concretos. El primero es la existencia de una plataforma web desplegable dentro del dominio de la Universidad de Antioquia y accesible desde navegador, tanto en computador como en dispositivo móvil [12]. El segundo es la integración de series temporales con filtros y normalización [1]. El tercero es la publicación contextualizada por estación. El cuarto es la incorporación de material audiovisual asociado a ciertas estaciones.

Desde un punto de vista sistémico, el resultado más importante puede escribirse como la

capacidad de construir una función de consulta

$$Q : (E_m, \mathcal{U}_m, \tau) \mapsto \mathcal{R},$$

donde E_m es una estación seleccionada, \mathcal{U}_m el subconjunto de variables de interés, τ una ventana temporal y \mathcal{R} el conjunto de salidas devueltas por la plataforma, incluyendo gráfico, metadatos y, cuando aplique, videos. Esta formulación resume el valor funcional del sistema: transformar una solicitud de consulta ambiental en una respuesta estructurada y navegable.

GeoVAM no se limita a visualizar datos aislados. La plataforma ubica estaciones, muestra sus variables, enlaza su contexto, organiza sus videos y articula distintos subsistemas de medición. En ese sentido, opera como una infraestructura digital de observación ambiental.

1.9. Discusión

GeoVAM ocupa una posición particular dentro del volumen porque no representa únicamente un desarrollo vertical, centrado en resolver un solo problema técnico, sino una solución transversal que articula varios. En capítulos como SUCA, CorrentómetroBT o SICUANU, puede identificarse con relativa facilidad un objeto técnico concreto: una aplicación móvil, un correntómetro o un sistema de análisis de nubosidad. En GeoVAM, en cambio, el objeto técnico principal es la articulación entre esos desarrollos.

La plataforma web no se entiende completamente si se la reduce a un código fuente y a unas vistas HTML. Su importancia reside en servir como superficie común para trabajos como los de Cristian Quintana, John Fernando Ramírez, Sara Ospina, Julián Muñoz y otros desarrollos que fueron organizando piezas de la red SIMEVAM. Por eso, la identificación de estos aportes resulta necesaria para mantener la trazabilidad del proyecto.

También debe señalarse una limitación documental. Los materiales compartidos permiten reconstruir con bastante solidez la finalidad, las vistas, los módulos y la inserción institucional de GeoVAM, pero no exponen todavía una memoria completa del código ni una batería formal de métricas de desempeño del sitio. Por esta razón, el capítulo se concentra en describir la arquitectura, la función y la lógica de integración sin atribuir detalles algorítmicos que no estén suficientemente documentados.

1.10. Conclusiones

GeoVAM es la plataforma web articuladora del ecosistema SIMEVAM, diseñada para publicar y hacer navegables los datos de variables ambientales medidos en tiempo real por estaciones y subsistemas asociados. Su aporte no radica únicamente en desplegar gráficos, sino en convertir una red de medición distribuida en una red de información accesible.

La arquitectura funcional del sistema puede leerse como una cadena que parte de estaciones y sensores, pasa por servicios de transmisión y almacenamiento, y culmina en una interfaz web compuesta por mapa, tabla, gráficos, páginas de detalle y repositorio de videos. Esta estructura permite formular GeoVAM como una función de integración multivariable y multifuente, adecuada para el contexto de monitoreo ambiental de la ciénaga de Ayapel y extensible a otros escenarios.

GeoVAM es además resultado de una construcción colectiva. En su consolidación deben reconocerse explícitamente los aportes de trabajos y desarrollos vinculados a SIMEVAM, entre ellos los asociados a Cristian Alexis Quintana Valencia, John Fernando Ramírez Gómez, Sara Ospina y Julián Muñoz, así como otros módulos que amplían la oferta instrumental y funcional del ecosistema, como SIREMEVAM, SUCA, CorrentómetroBT y SICUANU.

Desde el punto de vista institucional, GeoVAM expresa con claridad la orientación del seminario GeoLimna–GEPAR: problemas reales traducidos en sistemas medibles, algoritmos verificables y herramientas utilizables. En este caso, la herramienta no mide directamente el ambiente, pero hace posible que lo medido se vuelva inteligible, compartible y acumulable.

Agradecimientos

Se agradece a la Universidad de Antioquia y a los grupos de investigación GeoLimna y GEPAR por el entorno académico, técnico e institucional que hizo posible la construcción del ecosistema SIMEVAM y de la plataforma GeoVAM. Se reconoce además la contribución de los estudiantes y desarrolladores vinculados a los distintos subproyectos integrados o articulados a la plataforma, en especial los trabajos asociados a Cristian Alexis Quintana Valencia, John Fernando Ramírez Gómez, Sara Ospina y Julián Muñoz.

Referencias

- [1] GeoVAM, *GeoVAM: Manual de usuario*, Documento interno de los grupos GeoLimna y GEPAR, Universidad de Antioquia, Medellín, Colombia, 2021.
- [2] GeoVAM, *GeoVAM: Manual técnico*, Documento interno de los grupos GeoLimna y GEPAR, Universidad de Antioquia, Medellín, Colombia, 2021.
- [3] Universidad de Antioquia, «Sistema de registro y seguimiento de variables ambientales ciénaga de Ayapel (SIMEVAM), Acta de registro y de propiedad intelectual, Proyecto 2020-37870,» Universidad de Antioquia, Medellín, Colombia, Acta institucional, 5 de abr. de 2021.
- [4] Universidad de Antioquia, «Acta de inicio de trabajo de grado, SIMEVAM II. Sistema de medición de variables ambientales II,» Medellín, Colombia, Acta institucional, 14 de dic. de 2021.
- [5] C. A. Quintana Valencia, «Sistema remoto para la medición y registro de variables ambientales,» Trabajo de grado profesional, Universidad de Antioquia, Medellín, Colombia, 2022.
- [6] D. S. Fernández Mc Cann, F. d. J. Vélez Macías y N. J. Aguirre Ramírez, «Solicitud para asignación de puntos por desarrollo de software, SUCA - Aplicación móvil de sistema de medición superficial cautiva,» Universidad de Antioquia, Medellín, Colombia, Documento institucional, 13 de dic. de 2021.
- [7] Grupos Gepar y Geolimna, *SUCA: Hardware receptor y emisor de datos para sistema de medición superficial cautiva, Planos de diseño electrónico y documentación técnica*, Universidad de Antioquia, Medellín, Colombia, 2021.
- [8] F. d. J. Vélez Macías y D. S. Fernández Mc Cann, «Certificación de uso del software SUCA, Certificación institucional de uso por parte de los grupos GEOLIMNA y GEPAR,» Universidad de Antioquia, Medellín, Colombia, Certificación institucional, 9 de dic. de 2021.
- [9] Grupo GEPAR y Grupo GeoLimna, *CorrentómetroBT: Software para transmisión y recepción de datos de velocidad medida por sensor, Manual técnico*, Universidad de Antioquia, Medellín, Colombia, 2020.
- [10] Grupo GEPAR y Grupo GeoLimna, *CorrentómetroBT: Correntómetro con conexión Bluetooth a aplicación móvil, Manual de usuario*, Universidad de Antioquia, Medellín, Colombia, 2020.

- [11] D. S. Fernández Mc Cann, N. J. Aguirre Ramírez, F. d. J. Vélez Macías, J. F. Ramírez Gómez y M. Arroyave Cañas, *CorrentómetroBT: Sistema para la estimación de la velocidad de flujo de agua*, Registro de soporte lógico–software, Dirección Nacional de Derecho de Autor, Libro–Tomo–Partida 13-78-260, fecha de registro: 05-mar-2020, 2020.
- [12] J. Ríos, *GEOVAM*, <https://geovam.udea.edu.co/>, Sitio web institucional de la Universidad de Antioquia, 2021.

Capítulo 2

Clasificación de esquejes de crisantemo mediante visión por computador y aprendizaje automático

Autores

Juan Guillermo Pavas Henao¹; Dagoberto Castro Restrepo²; David Fernández Mc Cann³.

Resumen

EIVEN, *Esquejes de Crisantemo Identificados por sus Venas*, es un software de visión por computador orientado a la clasificación de esquejes de crisantemo según su calidad, mediante un flujo integrado de extracción de características, entrenamiento de modelos y predicción sobre nuevas muestras. El desarrollo responde a una necesidad del sector floricultor: reducir la subjetividad en la evaluación del estado de desarrollo de la hoja durante los procesos de recolección y control de calidad de esquejes, etapa relevante para lograr un enraizamiento homogéneo y una propagación eficiente. El sistema integra procesamiento digital de imágenes para segmentar venas, pecíolo, hoja y lóbulos, y aprendizaje automático para construir clasificadores capaces de discriminar estados de desarrollo foliar. En su validación experimental, realizada sobre 650 imágenes del cultivar *Baltica*, se defi-

¹Ingeniero, Maestría en Ingeniería.

²Profesor, Universidad Católica de Oriente.

³Profesor, Departamento de Ingeniería Electrónica y Telecomunicaciones.

^{1,2,3}Universidad de Antioquia. Grupos de investigación GEPAR y GeoLimna, en articulación con la Universidad Católica de Oriente.

nieron escenarios de entrenamiento basados en subconjuntos de rasgos morfológicos. Los resultados mostraron exactitudes de hasta 92 % y evidenciaron que los rasgos de lóbulos fueron los más informativos cuando se evaluaron de forma aislada. Este capítulo presenta el desarrollo, formaliza su arquitectura metodológica, describe su estructura de software y discute su alcance como herramienta para el sector floricultor y para nuevas líneas de investigación en analítica visual aplicada a agricultura de precisión.

Palabras clave: visión por computador; aprendizaje automático; esquejes de crisantemo; morfología foliar; floricultura; clasificación.

2.1. Introducción

En la propagación comercial del crisantemo, una parte importante del éxito agronómico depende de la selección inicial del esqueje que será llevado al banco de enraizamiento. Esa decisión, ejecutada de manera rutinaria por recolectores e inspectores de calidad, suele apoyarse en criterios visuales y en experiencia práctica. Cuando el criterio depende de la percepción humana sobre el desarrollo de la hoja, aparece una fuente de variabilidad difícil de controlar a gran escala.

EIVEN fue desarrollado para clasificar esquejes de crisantemo de acuerdo con su calidad mediante tareas de extracción de características, entrenamiento y clasificación, combinando procesamiento digital de imágenes y aprendizaje automático [1, 2, 3]. Su aporte principal consiste en transformar un problema agronómico en una estructura técnica medible, reproducible y transferible.

El desarrollo parte de una hipótesis concreta: el estado de desarrollo de la hoja no depende únicamente de su área. Existen rasgos morfológicos adicionales, como venación, pecíolo y lóbulos, que describen con mayor detalle el estado funcional del esqueje y que pueden cuantificarse mediante visión por computador [4, 5, 6]. Con base en esa hipótesis, el sistema organiza un flujo que va desde la imagen hasta la decisión clasificatoria.

El desarrollo cuenta con registro ante la Dirección Nacional de Derecho de Autor, certificación de uso, manual de usuario, manual técnico, tesis de maestría y producción académica derivada [2, 3, 4, 5, 6, 7, 8]. Esa trazabilidad documental lo sitúa como un producto de investigación aplicada y desarrollo tecnológico dentro del trabajo realizado entre universidad, grupos de investigación y sector floricultor.



Figura 2.1: Montaje de adquisición de imágenes empleado para la captura controlada de los esquejes y la generación del conjunto experimental de EIVEN.

2.2. Planteamiento del problema

El problema que da origen al desarrollo puede formularse de la siguiente manera: dado un conjunto de esquejes de crisantemo, se requiere determinar automáticamente su calidad con base en el estado de desarrollo de la hoja, de forma que la clasificación resultante sea útil para decidir su idoneidad en el proceso de propagación.

En términos formales, se trata de una tarea de clasificación multiclase. Sea

$$\mathcal{I} = \{f_1, f_2, \dots, f_N\}$$

el conjunto de imágenes de entrada, donde cada $f_i(x, y)$ representa la imagen de un esqueje o, con mayor precisión, de la hoja de interés del esqueje. A cada muestra debe asociarse una clase

$$y_i \in \mathcal{C} = \{\text{inmaduro, ideal, maduro}\},$$

de acuerdo con el estado de desarrollo foliar.

La clase y_i no se observa directamente en la imagen. Debe inferirse a partir de un conjunto de rasgos medibles. Por ello, el sistema construye para cada imagen f_i un vector de

características

$$\mathbf{x}_i \in \mathbb{R}^d,$$

obtenido a partir de la morfología de la hoja. En este caso, el vector integra información de venas, pecíolo, limbo y lóbulos [2, 3, 4].

La función de clasificación buscada puede expresarse de forma abstracta como

$$g : \mathbb{R}^d \rightarrow \mathcal{C},$$

donde $g(\mathbf{x}_i) = y_i$. El problema presenta al menos tres dificultades principales. La primera es agronómica: el criterio convencional de “hoja desarrollada” suele apoyarse en observación visual y no en una medición objetiva. La segunda es morfológica: el desarrollo foliar no depende de una sola magnitud, sino de una combinación de formas y proporciones. La tercera es computacional: esas estructuras deben segmentarse de forma robusta sin destruir el esqueje y sin desarticular el flujo productivo.

2.3. Motivación agronómica y novedad del desarrollo

La solicitud institucional del software establece que el desarrollo responde a una necesidad concreta del sector floricultor: disponer de un instrumento más preciso que el juicio visual para establecer el desarrollo de la hoja y, con ello, apoyar la identificación de esquejes ideales [1]. Aunque la longitud del tallo puede verificarse con guías físicas de corte, el criterio sobre desarrollo foliar sigue siendo subjetivo y, por tanto, susceptible de error.

La novedad del desarrollo consiste en proponer que el estado de desarrollo de la hoja puede inferirse no solo mediante el área foliar, sino también mediante rasgos morfológicos de venas, pecíolo y lóbulos. Esta propuesta cubre un vacío identificado en la solicitud de software, en la tesis y en los manuscritos derivados: existen antecedentes sobre venación, sobre lóbulos o sobre clasificación varietal, pero no se había trabajado de forma conjunta la contribución de esos rasgos al problema específico de calidad de esquejes de crisantemo en la etapa previa al enraizamiento [1, 4, 5, 6].

El procedimiento además es no destructivo. A diferencia de otros trabajos basados en hojas separadas del tallo, aquí se trabaja con imágenes de hojas aún unidas al esqueje, preservando la utilidad posterior del material vegetal [4, 6]. Esta decisión vincula la propuesta con el entorno real de producción y refuerza su orientación hacia la transferencia.

2.4. Arquitectura general del método

La tesis de maestría y el artículo derivado convergen en una estructura metodológica compuesta por dos grandes etapas: procesamiento digital de imágenes y aprendizaje automático [4, 5, 6]. Para este capítulo, dicha estructura puede expresarse como una arquitectura de seis fases:

$$f_i(x, y) \xrightarrow{\mathcal{P}} \mathbf{x}_i \xrightarrow{\mathcal{K}} \hat{y}_i^{(0)} \xrightarrow{\mathcal{S}} \hat{y}_i, \quad (2.1)$$

donde \mathcal{P} denota la etapa de procesamiento y extracción de características, \mathcal{K} la etapa de agrupamiento no supervisado para etiquetado inicial y \mathcal{S} la etapa de clasificación supervisada.

De manera más detallada, la fase \mathcal{P} se descompone en:

$$\mathcal{P} = \mathcal{P}_V \circ \mathcal{P}_P \circ \mathcal{P}_H \circ \mathcal{P}_L \circ \mathcal{E},$$

donde \mathcal{P}_V representa la segmentación de venas, \mathcal{P}_P la segmentación de pecíolo, \mathcal{P}_H la segmentación de hoja, \mathcal{P}_L la identificación de lóbulos y \mathcal{E} la extracción de características.

El resultado de esa etapa es un conjunto de vectores

$$X = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N\},$$

a partir del cual se construye el dataset para aprendizaje. Luego, el modelo K-means define una partición

$$X \mapsto \{C_1, C_2, C_3\},$$

que se interpreta como estados inmaduro, ideal y maduro según los valores de área y perímetro de la hoja. Finalmente, distintos modelos supervisados buscan una función

$$g_\theta(\mathbf{x}) = \hat{y},$$

donde θ representa el conjunto de hiperparámetros ajustados por validación cruzada.

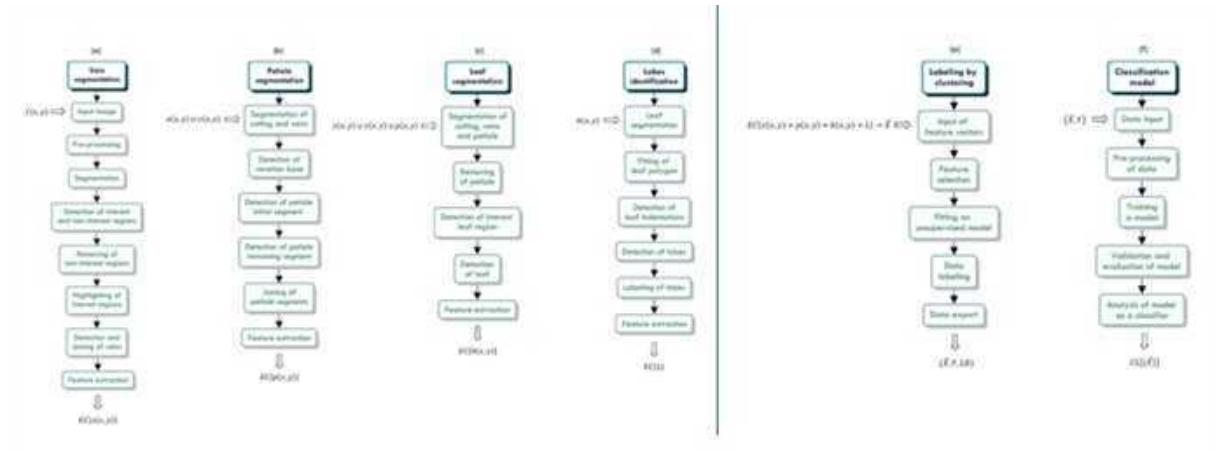


Figura 2.2: Flujo general del método, organizado en una etapa de procesamiento de imágenes y una etapa posterior de clasificación.

2.5. Adquisición de datos y representación morfológica

El sistema fue validado sobre un conjunto de 650 imágenes de hojas de esquejes del cultivar *Chrysanthemum × morifolium* ‘Baltica’ [4, 5, 6]. Las muestras provenían del proceso de control de calidad de la empresa Flores El Capiro S.A., y las imágenes fueron capturadas en un montaje de iluminación controlada con caja plástica, láminas acrílicas, módulos LED y un teléfono Xiaomi Redmi 6A con cámara de 13 MP [4, 5].

Debido a que un esqueje puede contener más de una hoja, el método selecciona la hoja de mayor área como hoja de interés. Sobre ella define cuatro componentes morfológicos:

$$H = \text{hoja}, \quad P = \text{pecíolo}, \quad V = \text{venas}, \quad L = \text{lóbulos}.$$

A partir de esos componentes se calculan rasgos básicos de área a , perímetro p , longitud l y diámetro d , además de coloración y simetría cuando corresponde [2, 3]. El vector de características puede representarse como

$$\mathbf{x}_i = [\mathbf{x}_i^{(V)}, \mathbf{x}_i^{(P)}, \mathbf{x}_i^{(H)}, \mathbf{x}_i^{(L)}],$$

donde cada bloque corresponde a una parte morfológica.

Entre los rasgos medidos se incluyen área, perímetro, longitud y diámetro de venas, pecíolo y hoja, así como longitudes de lóbulos superiores e inferiores en ambos laterales [4]. Esta representación descompone la hoja en subestructuras con significado morfológico y funcional, en lugar de reducirla a una silueta global.

2.6. Procesamiento digital de imágenes

La primera etapa del método consiste en transformar una imagen de un esqueje en un conjunto de objetos segmentados y medibles. Para ello se emplean técnicas clásicas de procesamiento digital de imágenes, combinadas con operaciones morfológicas, detección de contornos, esqueletización y análisis geométrico [3, 4, 6].

En la fase de preprocesamiento, la imagen $f(x, y)$ se transforma a distintos espacios de color para facilitar la separación entre fondo y objeto, así como para resaltar la venación. La tesis reporta el uso del componente C del espacio CMYK para remover el fondo y del componente L del espacio HSL, suavizado con filtro bilateral, para mejorar la visibilidad de venas [4, 6]. A partir de allí se definen imágenes intermedias que permiten umbralización global y adaptativa:

$$e(x, y) = \begin{cases} 1, & f'_1(x, y) > t, \\ 0, & \text{en otro caso,} \end{cases} \quad b(x, y) = \begin{cases} 1, & f'_2(x, y) > T(x, y), \\ 0, & \text{en otro caso.} \end{cases}$$

La imagen $e(x, y)$ representa el recorte binario del esqueje, mientras que $b(x, y)$ destaca regiones asociadas a la venación. Posteriormente, mediante detección de contornos, transformada de Hough probabilística y operaciones aritméticas y lógicas, se obtiene la segmentación de venas

$$v(x, y) = \mathcal{S}_V(b(x, y), e(x, y)).$$

Con $v(x, y)$ como referencia geométrica, el método localiza el pecíolo mediante perfiles de diámetro y seguimiento de regiones estrechas conectadas a la base de la venación:

$$p(x, y) = \mathcal{S}_P(e(x, y), v(x, y)).$$

Luego, la hoja se obtiene retirando el pecíolo del esqueje y reteniendo la región que coincide espacialmente con la venación:

$$h(x, y) = \mathcal{S}_H(e(x, y), p(x, y), v(x, y)).$$

Finalmente, sobre $h(x, y)$ se identifican lóbulos por medio de poligonización aproximada, casco convexo y defectos de convexidad:

$$L = \mathcal{I}_L(h(x, y)).$$

Esta cadena de segmentación conserva interpretabilidad geométrica y permite relacionar las decisiones computacionales con estructuras morfológicas observables.

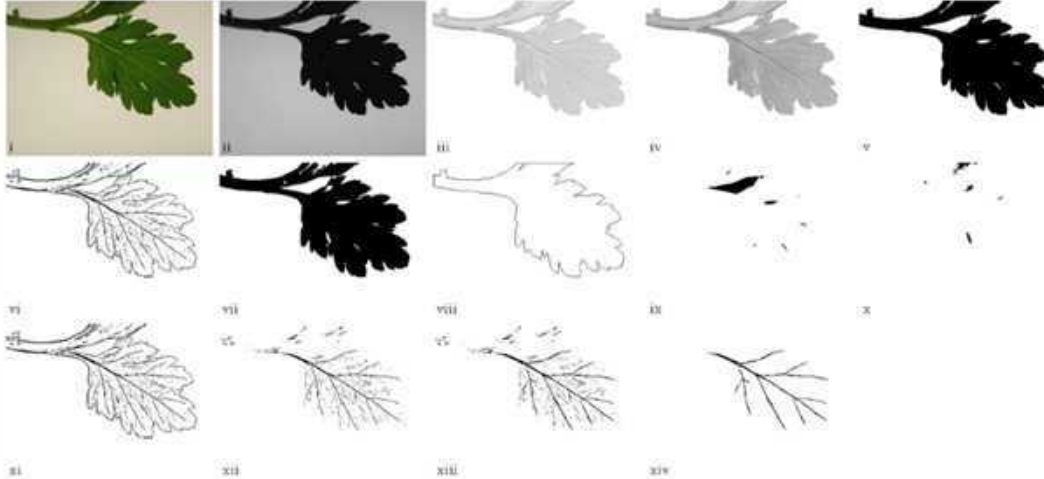


Figura 2.3: Ejemplo de segmentación de las partes morfológicas de la hoja, incluyendo venas, pecíolo, limbo y lóbulos detectados.

2.7. Definición de características y construcción del dataset

Una vez segmentadas las partes morfológicas, el sistema extrae las características que alimentan la etapa de aprendizaje automático. La definición base de los rasgos aparece en ambos manuales: área, perímetro, longitud, diámetro, coloración RGB y porcentaje de simetría [2, 3]. La tesis y el draft refinan esta representación para la experimentación científica, concentrándose en rasgos geométricos de venas, pecíolo, hoja y lóbulos [4, 6].

Si V , P , H y L representan respectivamente las partes morfológicas detectadas, el vector de características puede resumirse como

$$\mathbf{x}_i = [a_V, p_V, l_V, d_V, a_P, p_P, l_P, d_P, a_H, p_H, l_H, d_H, l_{LS}, l_{LSI}, l_{LII}, l_{LSD}, l_{LID}]_i.$$

En el software, además, el procesado por bloque permite exportar dichos rasgos en archivos planos CSV, línea por línea, una por imagen, junto con información adicional como número de venas secundarias, terciarias, ramificaciones y puntos finales, así como archivos de log para imágenes que no pudieron procesarse correctamente [2]. Esta capacidad conecta el procesamiento de imágenes con la experimentación reproducible mediante aprendizaje automático.

Si el conjunto de muestras exitosamente procesadas es N_s , el dataset queda representado por una matriz

$$X \in \mathbb{R}^{N_s \times d},$$

donde d es el número de características utilizadas en el escenario de entrenamiento correspondiente.

2.8. Aprendizaje automático: etiquetado y clasificación

La estrategia de aprendizaje automático se organizó en dos momentos. El primero consistió en etiquetar las muestras mediante agrupamiento no supervisado. El segundo, en entrenar clasificadores supervisados usando las etiquetas obtenidas [4, 5, 6].

2.8.1. Etiquetado por clustering

Dado que el criterio agronómico inicial de desarrollo foliar se relaciona con el tamaño de la hoja, el método utilizó área y perímetro del limbo para ajustar un modelo K-means con tres clústeres:

$$K = 3, \quad C = \{C_1, C_2, C_3\}.$$

Esos grupos se interpretaron como:

$$C_1 \rightarrow \text{inmaduro}, \quad C_2 \rightarrow \text{ideal}, \quad C_3 \rightarrow \text{maduro}.$$

Si $\mathbf{z}_i = [a_H, p_H]_i$ es el subvector de área y perímetro de la hoja, entonces el proceso de etiquetado busca

$$\min_{\{\boldsymbol{\mu}_k\}_{k=1}^3} \sum_{i=1}^{N_s} \min_{k \in \{1,2,3\}} \|\mathbf{z}_i - \boldsymbol{\mu}_k\|_2^2,$$

donde $\boldsymbol{\mu}_k$ es el centroide del clúster k . Esta formulación dio origen al conjunto de etiquetas con las que luego se entrenaron los clasificadores.

2.8.2. Clasificación supervisada

Con el dataset etiquetado, se entrenaron modelos de Regresión Logística (LR), Máquinas de Soporte Vectorial (SVM), Bosques Aleatorios (RF) y k -vecinos más cercanos (KNN)

[4, 5, 6]. De forma general, el problema consiste en estimar una función

$$g_{\theta} : \mathbb{R}^d \rightarrow \mathcal{C},$$

donde θ recoge los hiperparámetros ajustados mediante validación cruzada.

El entrenamiento se realizó usando una partición de 75 % para entrenamiento y validación y 25 % para prueba, con estrategias para tratar el desbalance de clases, incluyendo *class weights* y SMOTE según el modelo [4, 6]. Esta decisión fortaleció la evaluación y evitó que el desempeño reportado dependiera de una corrida aislada.

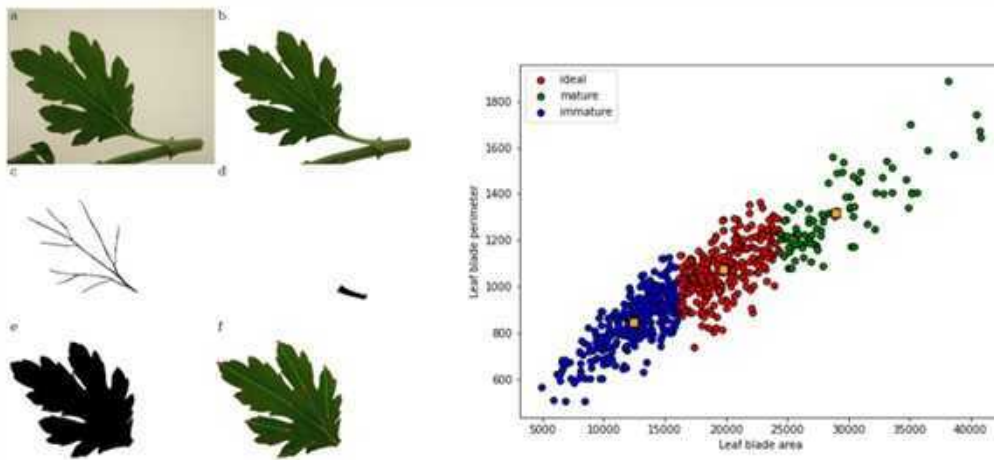


Figura 2.4: Agrupamiento inicial por K-means usando área y perímetro de la hoja para definir las clases inmaduro, ideal y maduro.

2.9. Arquitectura de software e implementación

Además de la contribución metodológica, el desarrollo tiene valor como software estructurado. El manual de usuario organiza la aplicación en tres módulos principales: extracción de características, entrenamiento y predicciones [2]. El manual técnico documenta requerimientos, dependencias, estructura de código, módulos de procesamiento digital de imágenes, utilidades y componentes de interfaz [3].

Desde el punto de vista técnico, el software fue desarrollado en Python 3.7.3 e integró bibliotecas como OpenCV, NumPy, Mahotas, Pandas, Scikit-learn, Matplotlib, Yellowbrick y PyQt [3]. Si se representa la aplicación como un sistema modular, puede describirse mediante la tupla

$$\mathcal{A} = (\mathcal{U}, \mathcal{P}, \mathcal{M}, \mathcal{R}), \quad (2.2)$$

donde:

\mathcal{U} = interfaz de usuario,

\mathcal{P} = módulos de procesamiento digital de imágenes,

\mathcal{M} = módulos de aprendizaje automático,

\mathcal{R} = rutinas de exportación y registro.

El manual técnico documenta módulos especializados como `venacion_pdi`, `peciolos_pdi`, `hojas_pdi`, `lobulos_pdi`, `morfologia`, `features`, `training` y `predicting` [3]. Esta modularidad favorece mantenibilidad, reutilización y evolución futura del sistema.

La interfaz expone al usuario la segmentación, las características extraídas, los archivos exportables, los resultados del entrenamiento y la carga de modelos para predecir nuevas muestras [2]. De este modo, la implementación convierte una metodología de investigación en una herramienta operable.



Figura 2.5: Interfaz principal del software y organización funcional de los módulos de extracción, entrenamiento y predicción.

2.10. Resultados

Los resultados reportados por la tesis, el paper draft y el trabajo de ExpoIngeniería son consistentes en lo esencial. El sistema logró segmentar las partes morfológicas de interés, construir el dataset de 650 muestras y evaluar varios escenarios de clasificación [4, 5, 6].

Cuando se utilizaron todas las características salvo área y perímetro del limbo, se obtu-

vieron exactitudes de 92 % para LR, 91 % para SVM y RF, y 87 % para KNN [4, 6]. Este resultado indica que el estado de desarrollo no depende exclusivamente del área visible de la hoja; otros rasgos retienen suficiente información para inferirlo.

Cuando se excluyeron por completo los rasgos del limbo, los porcentajes descendieron, pero se mantuvieron en niveles todavía informativos, entre 70 % y 81 % según el modelo [4, 5, 6]. Esto confirma que la combinación de venas, pecíolo y lóbulos describe una parte significativa del estado de desarrollo foliar.

Al evaluar partes morfológicas por separado, se encontraron tres hallazgos relevantes. Primero, las venas preservan una capacidad predictiva moderada, con desempeños alrededor de 63 % a 71 %. Segundo, el pecíolo aislado ofrece el rendimiento más bajo, cercano a 52 %–56 %. Tercero, los lóbulos fueron el mejor componente individual, con exactitudes entre 74 % y 80 % [4, 5, 6]. Estos resultados indican que la configuración lobular captura una señal morfológica estrechamente ligada al desarrollo de la hoja.

La tesis concluye además que SVM fue el clasificador más consistente, no solo por su desempeño, sino por requerir menos hiperparámetros que RF para alcanzar resultados comparables [4]. Esta observación orienta futuras implementaciones hacia modelos con buen equilibrio entre precisión y complejidad de ajuste.

2.11. Discusión

El desarrollo se diferencia de otros trabajos de clasificación vegetal por al menos cuatro rasgos. El primero es que aborda el problema de calidad de esquejes en etapa de propagación, no el de clasificación varietal en estados avanzados. El segundo es que opera con hojas aún unidas al esqueje, lo cual mantiene el carácter no destructivo del procedimiento. El tercero es que integra, en una misma representación, venas, pecíolo y lóbulos, componentes que usualmente aparecen aislados en la literatura. El cuarto es que culmina en un software funcional, documentado y registrado, y no solo en un flujo algorítmico experimental [1, 2, 3, 4, 7].

Los resultados también permiten una lectura más precisa. Que los lóbulos sean el mejor componente individual sugiere que la morfología del borde y su configuración espacial contienen información muy sensible al desarrollo foliar. Que las venas mantengan poder explicativo sin necesidad del área completa indica que la arquitectura interna de la hoja refleja su estado funcional. Que el pecíolo solo no sea suficiente, pero mejore al combinarse con venas, señala una relación complementaria más que autónoma.

La tesis también reporta limitaciones asociadas a oclusiones en lóbulos o discontinuidades

en la región de venación, que afectaron parte de la extracción morfológica [4]. Estas limitaciones delimitan el margen de mejora del sistema y orientan las proyecciones futuras, entre ellas el aumento de muestras, la incorporación de otros cultivares y la exploración de modelos más robustos, incluidos enfoques de aprendizaje profundo [4, 6].

2.12. Conclusiones

El desarrollo constituye un sistema sólido de visión por computador aplicado al control de calidad de esquejes de crisantemo. Su aporte principal consiste en haber formalizado un problema productivo real en una cadena técnica completa: adquisición no destructiva de imágenes, segmentación morfológica, extracción de rasgos, etiquetado por clustering, clasificación supervisada e implementación en software.

El capítulo mostró que el método no se apoya únicamente en el área de la hoja, sino en una representación morfológica más rica que involucra venas, pecíolo, hoja y lóbulos. Esta representación permitió construir modelos con desempeños altos, alcanzando hasta 92% de exactitud, y evidenció que los lóbulos fueron el componente individual más informativo, mientras que SVM fue el clasificador más consistente.

Desde la perspectiva institucional del seminario, el desarrollo integra investigación, desarrollo tecnológico y trazabilidad documental. Cuenta con tesis, manuales, certificación de uso, artículo publicado, manuscrito avanzado y registro DNDA. Por tanto, constituye una capacidad instalada para la universidad y una plataforma reusable para futuros problemas de analítica visual en floricultura y agricultura de precisión.

Agradecimientos

Se agradece a la Universidad de Antioquia, a la Universidad Católica de Oriente, a los grupos GEPAR y GeoLimna, y a la empresa Flores El Capiro S.A. por el acompañamiento académico, técnico y experimental que permitió el desarrollo del sistema. Se reconoce la contribución de Juan Guillermo Pavas Henao en la consolidación metodológica y experimental, así como el aporte de Dagoberto Castro Restrepo en la dimensión agronómica del problema y su articulación con el contexto productivo.

Referencias

- [1] D. S. Fernández Mc Cann, «Solicitud para asignación de puntos por desarrollo de software: EIVEN, Esquejes de Crisantemo Identificados por sus Venas,» Universidad de Antioquia, Medellín, Colombia, Documento institucional, 3 de abr. de 2023.
- [2] Universidad de Antioquia y Universidad Católica de Oriente, *EIVEN: Esquejes de Crisantemo Identificados por sus Venas. Manual de Usuario*, Universidad de Antioquia y Universidad Católica de Oriente, Medellín, Colombia, 2021.
- [3] Universidad de Antioquia y Universidad Católica de Oriente, *EIVEN: Esquejes de Crisantemo Identificados por sus Venas. Manual Técnico*, Universidad de Antioquia y Universidad Católica de Oriente, Medellín, Colombia, 2021.
- [4] J. G. Pavas Henao, «Computer vision method to determine the quality of Chrysanthemum cuttings through their leaf morphology,» Master's thesis, Universidad de Antioquia, Medellín, Colombia, 2023.
- [5] J. G. Pavas-Henao, D. Fernández-Mc Cann y D. Castro-Restrepo, «Method for analyzing morphological traits of chrysanthemum cuttings based on computer vision,» en *Engineering for Transformation*, Editorial EIA, 2022, págs. 451-458, ISBN: 978-628-95287-1-8.
- [6] J. G. Pavas-Henao y D. Fernandez-Mc Cann, «Computer vision method for classifying Chrysanthemum cuttings based on leaf developmental stage,» Draft de artículo, preprint fechado el 16 de marzo de 2026, 2026.
- [7] D. Fernández Mc Cann, D. Castro Restrepo y J. G. Pavas Henao, *EIVEN*, Registro de soporte lógico–software ante la Dirección Nacional de Derecho de Autor, Libro 13, Tomo 89, Partida 29, fecha de registro: 24-feb-2022, 2022.
- [8] D. S. Fernández Mc Cann, «Certificación de uso del software EIVEN por parte del grupo de investigación GEPAR,» Universidad de Antioquia, Medellín, Colombia, Certificación institucional, 3 de abr. de 2023.

Capítulo 3

Monitoreo multitemporal de coberturas del suelo en áreas de minería de oro aluvial mediante teledetección y minería de patrones secuenciales

Autores

Maura Melissa Herrera Ceferino¹; Nicolás Albornoz Lozano²; Esteban Escobar Chaverra³; Fabio de Jesús Vélez Macías⁴; Néstor Jaime Aguirre Ramírez⁵; David Stephen Fernández Mc Cann⁶.

Resumen

El seguimiento de la minería de oro aluvial en zonas ecológicamente sensibles exige herramientas capaces de integrar series históricas de imágenes satelitales, criterios de clasificación de coberturas y mecanismos de análisis multitemporal con bajo costo computacional y alta interpretabilidad. En este capítulo se presenta un desarrollo metodológico y compu-

¹Estudiante de Maestría en Ingeniería Ambiental.

²Estudiante de Ingeniería Electrónica y Telecomunicaciones.

³Estudiante de Ingeniería Electrónica y Telecomunicaciones.

⁴Profesor, Departamento de Ingeniería Ambiental y Sanitaria.

⁵Profesor, Departamento de Ingeniería Ambiental y Sanitaria.

⁶Profesor, Departamento de Ingeniería Electrónica y Telecomunicaciones.

^{1,2,3,4,5,6}Universidad de Antioquia. Grupos de investigación GeoLimna y GEPAR.

tacional orientado al monitoreo de cambios de cobertura del suelo en áreas intervenidas por minería aluvial en Ayapel, Córdoba, tomando como caso de estudio la microcuenca Quebradona, al sur del sistema cenagoso de Ayapel. El trabajo articula teledetección, procesamiento digital de imágenes, clasificación no supervisada y minería de patrones secuenciales para construir una herramienta que permite clasificar coberturas del suelo, cuantificarlas y estudiar su evolución temporal. La propuesta se apoya en imágenes Landsat corregidas a nivel atmosférico, en el uso de índices espectrales como SAVI y NDVI, en una clasificación basada en K-means y en un análisis multitemporal mediante el algoritmo SPADE. Además de la formulación metodológica, el capítulo documenta su materialización en un software operativo, registrado y documentado, orientado a simplificar el análisis de imágenes satelitales para usuarios del ámbito ambiental. Se discuten los resultados obtenidos, la capacidad del sistema para aproximarse al comportamiento histórico de la minería y la vegetación, y su valor como herramienta de apoyo a la gestión ambiental y al monitoreo de procesos de restauración ecológica.

Palabras clave: teledetección; minería de oro aluvial; Landsat; SAVI; K-means; análisis multitemporal; patrones secuenciales; restauración ecológica.

3.1. Introducción

El monitoreo de áreas mineras mediante teledetección se ha convertido en una necesidad técnica y política en regiones donde la transformación del paisaje ocurre de manera acumulativa, dispersa y, con frecuencia, al margen de la regulación estatal. En contextos como el de Ayapel, donde la minería aluvial ha alterado fuertemente la cobertura vegetal, los cauces y la estructura ecológica del territorio, la observación de campo resulta insuficiente para reconstruir la magnitud histórica del fenómeno. Se requiere una metodología que permita analizar el territorio como una serie temporal y no como una imagen aislada.

El desarrollo documentado en este capítulo responde a esa necesidad. Su propósito fue construir un sistema capaz de seguir el cambio de coberturas del suelo a través del tiempo, identificar patrones asociados a actividad minera y ofrecer una lectura operativa del territorio mediante una herramienta informática. El software asociado al proyecto fue concebido para clasificar coberturas del suelo en áreas mineras utilizando imágenes Landsat y para realizar análisis multitemporal de los cambios observados [1, 2, 3, 4].

La motivación territorial es precisa. La cuenca del río San Pedro y las microcuencas que alimentan la ciénaga de Ayapel han sufrido procesos intensos de deforestación y modificación del paisaje asociados a minería de aluvión. La situación es especialmente crítica en Quebradona, Escobillas y Trejos, donde la alteración del bosque de galería ha

tenido efectos directos sobre la estabilidad ecológica de la región [5, 6]. Dado que el sistema cenagoso de Ayapel fue declarado sitio Ramsar, el seguimiento de estos cambios responde tanto a intereses académicos como a exigencias de manejo ambiental.

El capítulo presenta el desarrollo en tres niveles. El primero es científico, porque parte de una pregunta sobre cómo seguir históricamente la transformación de coberturas en áreas mineras. El segundo es metodológico, porque diseña una cadena técnica basada en índices espectrales, clasificación y minería de patrones. El tercero es tecnológico, porque esa cadena se incorpora en un software con interfaz, manuales, ejecutable, rutas de exportación y registro de propiedad intelectual.

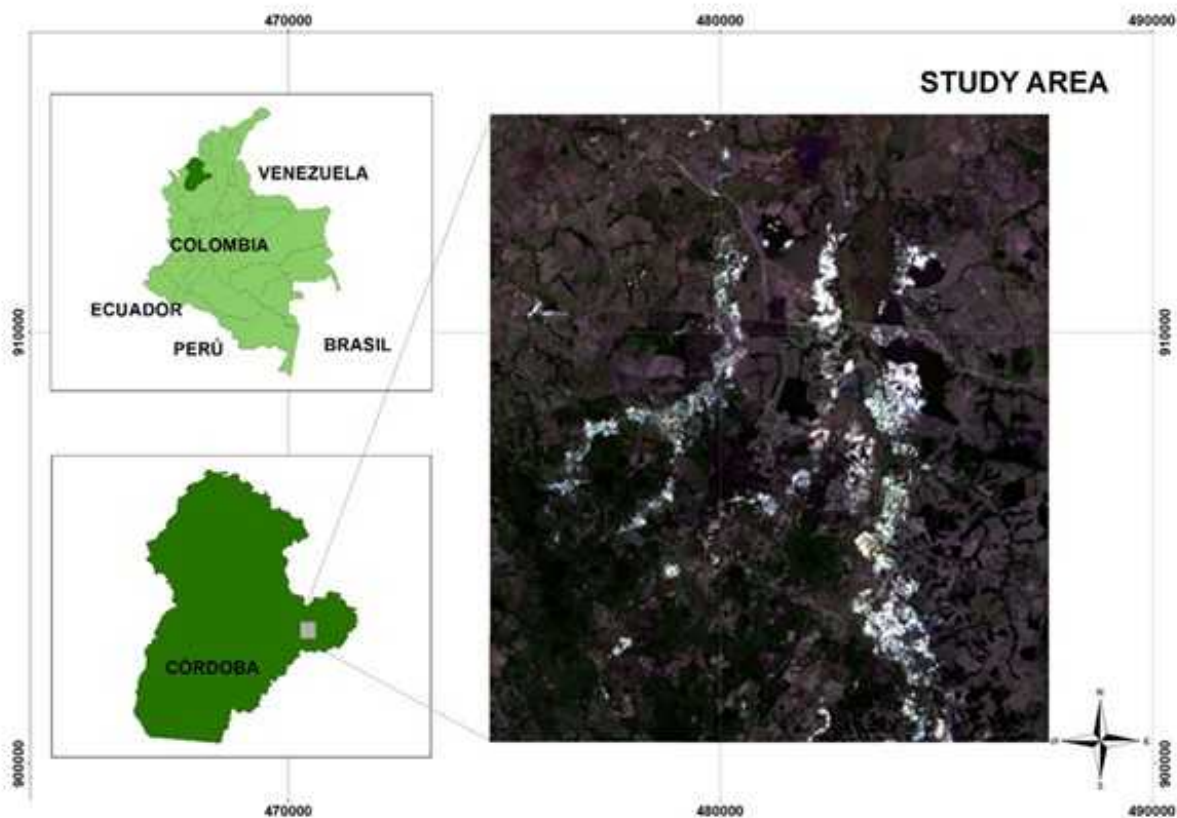


Figura 3.1: Área de estudio correspondiente a la microcuenca Quebradona, ubicada en el sistema cenagoso de Ayapel, y contexto territorial del problema abordado.

3.2. Problema y justificación

El problema abordado puede formularse así: dado un conjunto de imágenes satelitales de una misma región tomadas en fechas distintas, se requiere identificar y cuantificar

las coberturas del suelo asociadas a actividad minera y vegetación, y luego analizar su evolución temporal para inferir tendencias de degradación o recuperación.

Si se considera una serie temporal de imágenes

$$\mathcal{I} = \{I_{t_1}, I_{t_2}, \dots, I_{t_T}\},$$

donde cada I_{t_k} corresponde a una captura Landsat en el instante t_k , el objetivo consiste en construir, para cada imagen, una clasificación

$$C_{t_k}(i, j) \in \Omega,$$

donde Ω es el conjunto de clases de cobertura del suelo. En este trabajo, dicho conjunto incluye categorías como suelo desnudo, vegetación baja, vegetación media, vegetación alta, agua, nubes y suelo minero activo [3, 5, 6].

La necesidad de resolver este problema no es únicamente cartográfica. La actividad minera de aluvión modifica el mosaico espacial del territorio, elimina vegetación, altera cuerpos de agua y deja huellas cuya permanencia o reversión en el tiempo puede leerse como indicador indirecto de impacto y restauración. Por ello, una imagen aislada no basta. Se requiere una estructura analítica que permita seguir secuencias de cambio.

La documentación institucional del software indica además que el uso de imágenes satelitales brinda una posibilidad eficaz para monitorear cambios en tierra y agua, mientras que el desarrollo de software aporta simplicidad de análisis, bajo costo computacional y una aproximación operativa para entidades y usuarios ambientales [2]. El aporte del desarrollo no está solo en los algoritmos empleados, sino en su integración como herramienta utilizable.

3.3. Contexto metodológico y territorial

La propuesta se desarrolló sobre un polígono ubicado al sureste de la ciénaga de Ayapel, donde se encuentra la microcuenca Quebradona, una de las más afectadas por minería de oro aluvial. La tesis y el artículo asociado describen la zona como parte del macro-sistema de humedales y llanuras aluviales de La Mojana, con alta importancia ecológica e hidrológica [5, 6, 7].

El conjunto de datos provino de imágenes Landsat T1 descargadas desde Google Earth Engine, correspondientes a las misiones Landsat 4, 5 y 8. Las imágenes tenían preprocesamiento atmosférico tipo TOA y cubrían un intervalo entre 1985 y 2020 [5, 6, 7]. Esta

amplitud temporal convierte el problema en un análisis histórico del paisaje y no en un ejercicio puntual de clasificación.

Desde la perspectiva conceptual, la propuesta se ubica en el cruce entre teledetección y restauración ecológica. La minería aluvial elimina cobertura vegetal y transforma el patrón espacial del territorio; por ello, seguir la evolución de las coberturas permite aproximarse al deterioro y, en ciertos casos, a la revegetación posterior [6, 7]. Ese vínculo entre cobertura y restauración estructura buena parte del sentido del desarrollo.

3.4. Arquitectura general del desarrollo

La lógica del sistema puede resumirse como una secuencia de tres etapas:

$$\mathcal{S} = \mathcal{C} \circ \mathcal{P} \circ \mathcal{M},$$

donde \mathcal{M} representa la preparación multitemporal de imágenes de entrada, \mathcal{P} la fase de clasificación y postprocesado para cada fecha, y \mathcal{C} la etapa de análisis secuencial sobre la serie resultante.

De manera más explícita, para cada imagen I_t se construye primero una imagen clasificada preliminar basada en índices espectrales y agrupamiento:

$$I_t \xrightarrow{\text{SAVI} + \text{K-means}} C_t^{(0)}.$$

Luego, un postprocesado añade nuevas clases y corrige parcialmente la interpretación:

$$C_t^{(0)} \xrightarrow{\text{NDWI} + \text{red hídrica} + \text{nubes} + \text{reglas}} C_t.$$

Finalmente, sobre el conjunto ordenado

$$\{C_{t_1}, C_{t_2}, \dots, C_{t_T}\}$$

se aplica minería de patrones secuenciales para producir secuencias de transición por píxel:

$$\{C_{t_k}\}_{k=1}^T \xrightarrow{\text{SPADE}} \mathcal{Q},$$

donde \mathcal{Q} es el conjunto de secuencias frecuentes con sus valores de soporte y confianza.

Esta estructura distingue con claridad entre clasificación por fecha y lectura histórica del territorio. El sistema utiliza la primera como insumo de la segunda, sin confundir un mapa clasificado con un análisis multitemporal completo.

3 Monitoreo multitemporal de coberturas del suelo en áreas de minería de oro aluvial mediante teledetección y minería de patrones secuenciales

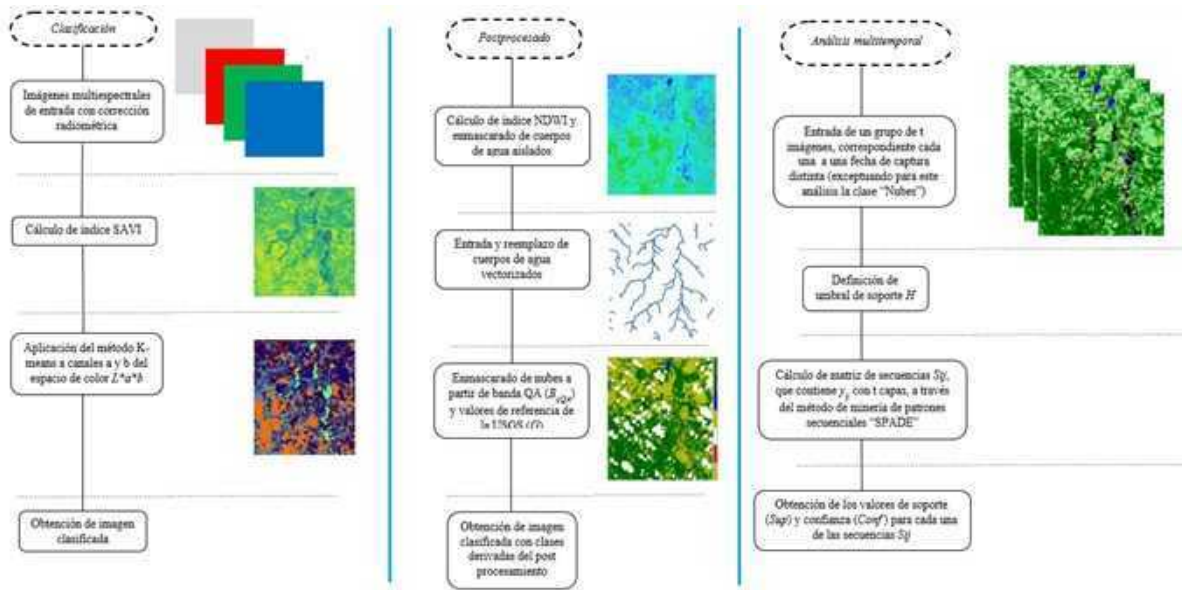


Figura 3.2: Fases principales del procesamiento satelital, incluyendo clasificación, post-procesado y análisis multitemporal.

3.5. Clasificación de coberturas

La primera etapa del método consiste en clasificar las coberturas del suelo a partir de las bandas Landsat. Para ello, el sistema calcula en primer lugar el índice SAVI, ampliamente utilizado cuando existe presencia de suelo desnudo y se desea compensar su efecto en la respuesta espectral de la vegetación [5, 6, 7].

El SAVI puede expresarse como

$$SAVI = \frac{(NIR - R)}{(NIR + R + L)}(1 + L),$$

donde NIR es la banda del infrarrojo cercano, R la banda roja y L un parámetro de ajuste del suelo. En este desarrollo se utilizó $L = 0,5$, valor coherente con una superficie medianamente expuesta [3, 6, 7].

Una vez obtenido el índice, la imagen se lleva al espacio de color Lab y se aplica el algoritmo K-means sobre componentes seleccionadas para agrupar píxeles con comportamiento similar. Formalmente, si $x_n \in \mathbb{R}^d$ representa el vector espectral o transformado asociado

al píxel n , K-means busca una partición en K grupos minimizando

$$J = \sum_{k=1}^K \sum_{x_n \in C_k} \|x_n - \mu_k\|^2,$$

donde μ_k es el centroide del clúster C_k . En este caso, la clasificación preliminar se usó para distinguir clases como suelo desnudo, vegetación baja, vegetación media y vegetación alta [3, 4, 6, 7].

La clasificación no supervisada inicial no se asume como resultado final. El sistema está diseñado para que el usuario, apoyado por la vista RGB, el SAVI y el NDVI, supervise la asignación semántica de los grupos resultantes y complete una clasificación más interpretable [3]. Esta decisión combina automatización con criterio experto, algo especialmente útil en aplicaciones ambientales donde la interpretación del territorio no es puramente mecánica.

3.6. Postprocesado y construcción de nuevas clases

El postprocesado constituye una etapa central del desarrollo, porque la clasificación inicial no basta para distinguir adecuadamente cuerpos de agua, nubes o zonas de minería activa. Para ello se introduce, entre otros elementos, el índice NDWI, definido como

$$\text{NDWI} = \frac{G - \text{NIR}}{G + \text{NIR}},$$

donde G es la banda verde y NIR la banda del infrarrojo cercano. Su propósito es resaltar cuerpos de agua abiertos y mejorar su separación frente a otras coberturas [6, 7].

A partir del NDWI se construye una máscara binaria de agua

$$W(i, j) = \begin{cases} 1, & \text{NDWI}(i, j) > \tau, \\ 0, & \text{en otro caso,} \end{cases}$$

que luego se complementa con una red hídrica vectorizada de ríos y quebradas. Sobre esa base, el sistema detecta además cuerpos de agua aislados susceptibles de corresponder a piscinas mineras, y utiliza reglas geométricas y de vecindad para diferenciar áreas de suelo minero activo alrededor de ellas [3, 4, 6, 7].

La identificación de nubes se hace usando la banda de calidad QA y valores de referencia

definidos por la USGS. De esta forma, el sistema produce una imagen final clasificada

$C_t(i, j) \in \{\text{suelo desnudo, veg. baja, veg. media, veg. alta, agua, nubes, suelo minero activo}\}$.

Esta etapa incorpora conocimiento específico del problema. El sistema distingue entre suelo desnudo general y suelo minero activo, y evita interpretar de forma equivalente regiones con respuestas espectrales similares pero con significado territorial distinto.

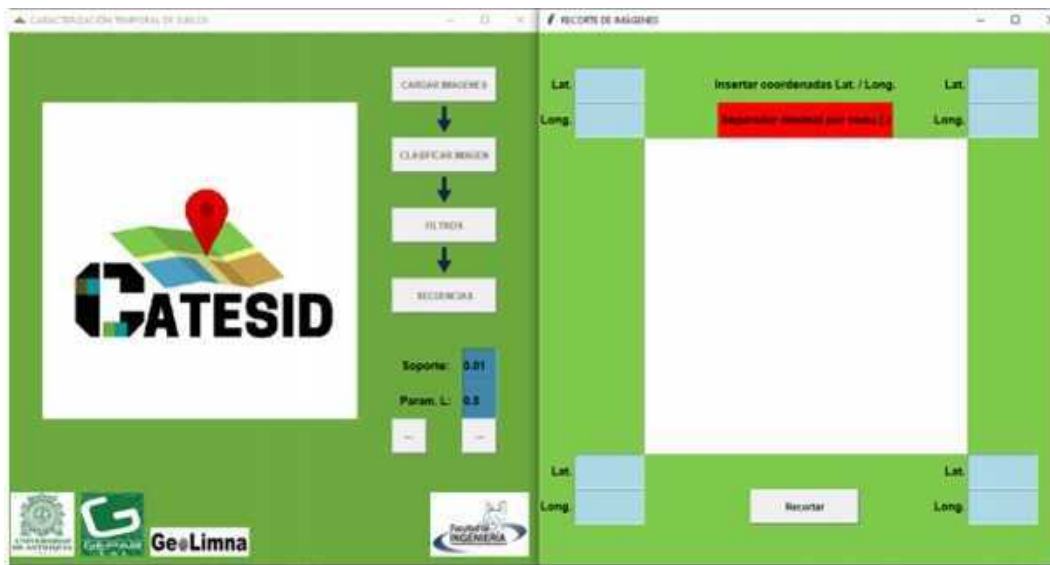


Figura 3.3: Interfaz de clasificación y apoyo visual al usuario durante la supervisión inicial del proceso.

3.7. Análisis multitemporal mediante patrones secuenciales

La etapa más distintiva del desarrollo es el análisis multitemporal. En lugar de limitarse a comparar dos fechas, el sistema trabaja con secuencias de coberturas por píxel, ordenadas cronológicamente. Si cada imagen clasificada C_t es una matriz $m \times n$, entonces para cada posición (i, j) puede definirse una secuencia

$$S_{ij} = (C_{t_1}(i, j), C_{t_2}(i, j), \dots, C_{t_T}(i, j)).$$

El conjunto de todas esas secuencias constituye una base transaccional espacial sobre la que se aplica el algoritmo SPADE. El objetivo es encontrar secuencias frecuentes de

transición entre clases, junto con sus medidas de soporte y confianza [3, 5, 6, 7]. Si σ es una secuencia dada, su soporte puede expresarse como

$$\text{Sup}(\sigma) = \frac{\#\{S_{ij} : \sigma \subseteq S_{ij}\}}{m n},$$

mientras que la confianza relativa a un antecedente α puede escribirse como

$$\text{Conf}(\alpha \rightarrow \beta) = \frac{\#\{S_{ij} : \alpha \cup \beta \subseteq S_{ij}\}}{\#\{S_{ij} : \alpha \subseteq S_{ij}\}}.$$

Estas expresiones formalizan la lectura temporal del territorio. El sistema no solo mide cuánto suelo minero existe, sino qué trayectorias de cambio siguen los píxeles. Por ejemplo, una secuencia del tipo

suelo minero activo \rightarrow vegetación baja \rightarrow vegetación media

puede interpretarse como una transición asociada a recuperación de cobertura. En cambio,

vegetación media \rightarrow suelo desnudo \rightarrow suelo minero activo

puede leerse como intensificación del proceso extractivo.

El valor del algoritmo radica en que revela patrones recurrentes del mosaico espacial y temporal del territorio, más allá de una comparación simple entre mapas.

3.8. Implementación en software

El desarrollo se concretó en un software ejecutable para Windows, documentado mediante manual de usuario, manual técnico e instrucciones de instalación. El sistema cuenta con una interfaz gráfica donde el usuario puede cargar imágenes, recortarlas, visualizar la escena RGB, clasificar, filtrar clases y ejecutar el análisis multitemporal [3, 4, 8].

Desde la perspectiva arquitectónica, el software puede representarse por la tupla

$$\mathcal{A} = (\mathcal{U}, \mathcal{R}, \mathcal{C}, \mathcal{F}, \mathcal{T}),$$

3 Monitoreo multitemporal de coberturas del suelo en áreas de minería de oro aluvial mediante teledetección y minería de patrones secuenciales

donde:

\mathcal{U} = interfaz de usuario,

\mathcal{R} = gestión de rásteres e índices espectrales,

\mathcal{C} = módulos de clasificación y postprocesado,

\mathcal{F} = módulos de filtrado y exportación de estadísticas,

\mathcal{T} = módulo de análisis temporal por secuencias.

El manual técnico muestra además que el programa fue desarrollado en Windows 10, usando Python vía Anaconda y bibliotecas como NumPy, Matplotlib, Tkinter, OpenCV, Rasterio, Fiona, Pandas, Geopandas, GDAL y `sklearn.cluster.KMeans` [4]. El manual de usuario evidencia que el sistema genera carpetas para resultados intermedios y finales, exporta archivos Excel con filtros y secuencias, y mantiene una lógica operativa clara para usuarios no necesariamente expertos en programación [3].

En consecuencia, el software no opera como un anexo instrumental, sino como un producto tecnológico con autonomía funcional y documental.

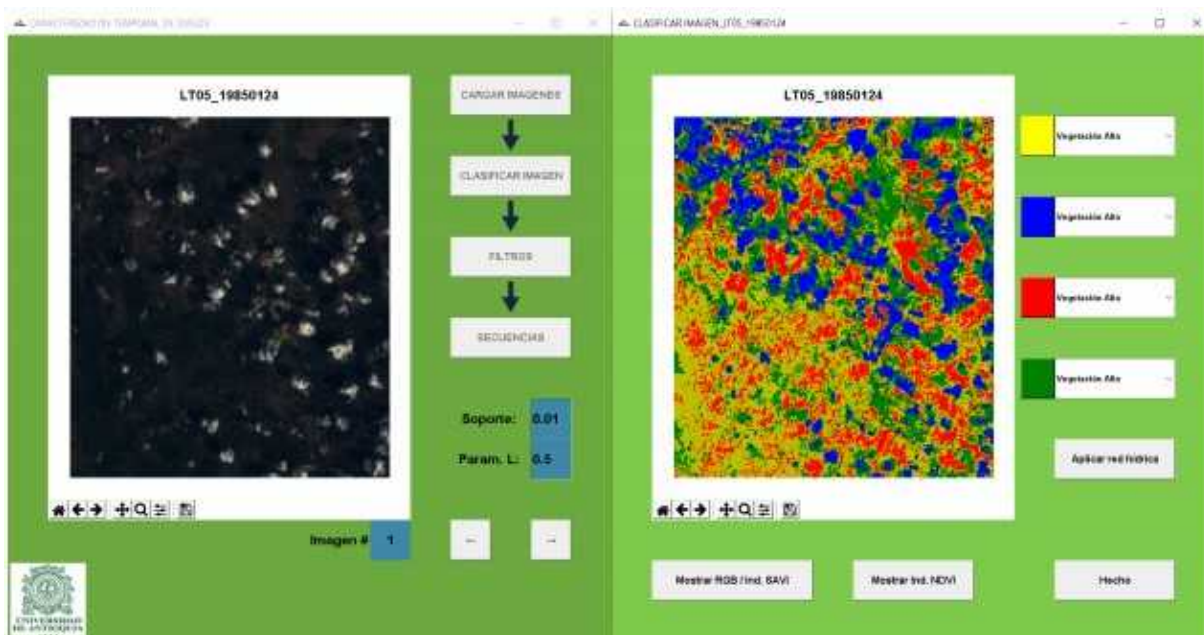


Figura 3.4: Ventana principal del sistema con los módulos de carga, clasificación, filtrado y análisis secuencial.

3.9. Resultados

Los resultados reportados por la tesis, el artículo y la ponencia muestran que la metodología logró una precisión general total de

$$74,37\% \pm 0,10,$$

calculada como el promedio de las precisiones generales obtenidas por imagen clasificada [5, 6, 7]. El coeficiente Kappa alcanzó un valor de

$$\kappa = 0,658,$$

interpretado como una buena clasificación [6, 7].

También se observó que las imágenes de Landsat 4 y 5 arrojaron resultados más modestos que Landsat 8, lo cual se relaciona con diferencias en resolución radiométrica y, por ende, con la capacidad de distinguir coberturas [5, 7]. La presencia de nubosidad generó además subestimaciones de ciertas coberturas, en especial en imágenes como la de 2013, mientras que en otros casos hubo sobreestimación del suelo desnudo o del suelo minero activo por similitud en niveles de reflectancia [5, 7].

En el plano territorial, los resultados permitieron reconstruir tendencias históricas. La documentación reporta un aumento acumulado de minería respecto a 1985, con un pico fuerte hacia 2010, así como una reducción de la vegetación media respecto al inicio del periodo, aunque con ciertos episodios de recuperación posterior [7]. Estos resultados muestran que el sistema convierte series Landsat en una lectura histórica del territorio, algo especialmente útil en contextos donde el inventario continuo en campo sería costoso y difícil de sostener.

3.10. Discusión

El principal valor del desarrollo radica en que propone una vía intermedia entre métodos de alta complejidad y análisis manuales de baja trazabilidad. Frente a enfoques de aprendizaje profundo que exigen grandes volúmenes de entrenamiento y mayor costo computacional, esta propuesta logra una aproximación útil con una arquitectura más interpretable y basada en información libre [2, 5]. Esa decisión responde de manera coherente al tipo de usuario y al objetivo de transferencia.

Otro aporte importante es la incorporación del análisis multitemporal por secuencias. Muchos estudios de cobertura comparan mapas entre fechas; aquí, en cambio, se estudian

trayectorias por píxel. Esto añade una capa interpretativa especialmente relevante para hablar de restauración ecológica, ya que permite distinguir entre permanencia, degradación y revegetación.

El desarrollo también delimita sus alcances con claridad. La clasificación no alcanza valores cercanos al 100 %, y las condiciones de nubosidad o similitud espectral entre clases afectan la precisión. La documentación plantea rutas de mejora consistentes, entre ellas preprocesado más robusto, mayor contraste radiométrico e incorporación futura de imágenes Sentinel de mayor resolución [5]. Estas proyecciones amplían la línea de trabajo en lugar de cerrarla.

3.11. Conclusiones

El desarrollo presentado en este capítulo demuestra que es posible construir una herramienta computacional para el seguimiento histórico de coberturas del suelo en áreas de minería aluvial, integrando imágenes Landsat, índices espectrales, clasificación no supervisada, reglas de postprocesado y minería de patrones secuenciales.

La propuesta no solo permitió clasificar coberturas y estimar cambios de vegetación y suelo minero en Ayapel, sino que además materializó esa metodología en un software registrado, documentado y utilizable por investigadores y actores del ámbito ambiental. En consecuencia, el producto constituye una plataforma metodológica y tecnológica de monitoreo territorial.

Desde la perspectiva del seminario GeoLimna–GEPAR, este capítulo muestra con claridad una línea de trabajo en la que problemas ambientales reales se traducen en cadenas técnicas medibles, algoritmos verificables y herramientas concretas. En este caso, el aporte consiste en construir una forma rigurosa de observar en el tiempo la huella territorial de la minería de aluvión.

Agradecimientos

Se agradece a la Universidad de Antioquia, a los grupos GeoLimna y GEPAR, y a todos los participantes del proyecto por la consolidación del desarrollo metodológico y computacional aquí presentado. Se reconoce de manera especial el aporte de Maura Melissa Herrera Ceferino al trabajo de investigación del que este software deriva, así como la contribución de Nicolás Albornoz Lozano y Esteban Escobar Chaverra al desarrollo del aplicativo, tal como se consigna en la documentación asociada [1, 5, 7].

Referencias

- [1] D. S. Fernández Mc Cann, F. d. J. Vélez Macías, M. M. Herrera Ceferino, N. Albornoz Lozano y E. Escobar Chaverra, *CATESID: Software para la caracterización temporal de suelos usando imágenes digitales*, Registro ante la Dirección Nacional de Derecho de Autor, Libro 13, Tomo 89, Partida 262, fecha de registro: 31-mar-2022, 2022.
- [2] D. S. Fernández Mc Cann y F. d. J. Vélez Macías, «Solicitud para asignación de puntos por desarrollo de software: CATESID, Caracterización temporal de suelos usando imágenes digitales,» Universidad de Antioquia, Medellín, Colombia, Documento institucional, 3 de abr. de 2023.
- [3] Universidad de Antioquia, *CATESID: Caracterización temporal de suelos usando imágenes digitales. Manual de usuario*, Universidad de Antioquia, Medellín, Colombia, 2021.
- [4] Universidad de Antioquia, *CATESID: Caracterización temporal de suelos usando imágenes digitales. Manual técnico*, Universidad de Antioquia, Medellín, Colombia, 2021.
- [5] M. M. Herrera, D. S. Fernández, F. d. J. Vélez y N. J. Aguirre, «Software for identifying alluvial gold mining areas, their characteristic coverages, and their multitemporal analysis,» en *Engineering for Transformation*, Trabajo presentado en IC EXPOI 2022, 2022.
- [6] M. M. Herrera Ceferino, D. S. Fernández Mc Cann y F. d. J. Vélez Macías, «Análisis del cambio multitemporal de coberturas del suelo en áreas mineras de oro aluvial en Ayapel Córdoba-Colombia, utilizando procesamiento digital de imágenes y minería de patrones secuenciales,» Borrador de artículo, 2022.
- [7] M. M. Herrera Ceferino, «Monitoreo de la restauración de la cobertura vegetal en áreas impactadas por minería de oro de aluvión, a partir de imágenes satelitales, en Ayapel Córdoba,» Tesis de Maestría en Ingeniería Ambiental, Universidad de Antioquia, Medellín, Colombia, 2022.
- [8] Universidad de Antioquia, *Instrucciones para abrir el software CATESID*, Universidad de Antioquia, Medellín, Colombia, 2021.

Capítulo 4

Laboratorio remoto para enseñanza y experimentación en redes de datos mediante control web de dispositivos

Autores

Juan Alejandro Ríos Pérez¹; Sebastián López Stan²; Yeison David Ballesteros³; Gustavo Andrés Marín Lopera⁴; David Stephen Fernández Mc Cann⁵.

Resumen

La interrupción del acceso presencial a un laboratorio no implica solo la pérdida de un espacio físico, sino también la limitación de una forma de aprendizaje basada en la interacción directa con dispositivos y topologías reales. En los cursos de redes de datos, el trabajo con routers, switches, terminales y errores de configuración constituye una parte esencial de la formación. Este capítulo presenta una plataforma web orientada al control remoto de dispositivos de red en un laboratorio académico, capaz de administrar inventario, crear prácticas, reservar recursos y lanzar sesiones de trabajo sobre equipos reales y virtualizados. La propuesta fue concebida para instituciones educativas y organiza la interacción

¹Estudiante de Ingeniería de Sistemas.

²Estudiante de Ingeniería de Sistemas.

³Estudiante de Ingeniería de Sistemas.

⁴Profesor y coordinador del laboratorio de Telemática.

⁵Profesor, Departamento de Ingeniería Electrónica y Telecomunicaciones.

^{1,2,3,4,5}Universidad de Antioquia. Grupo de investigación GEPAR.

según roles de administrador, profesor y estudiante, reproduciendo por software la lógica operativa de un laboratorio de redes. El desarrollo combina una arquitectura web basada en Node.js, frontend en HTML5 con Handlebars, base de datos MySQL, autenticación con integración externa, control remoto por terminal y conexión con infraestructura física y virtual del laboratorio. En este capítulo se describe el problema que dio origen al sistema, se formaliza su arquitectura funcional, se explica su diseño modular y se discute su valor como herramienta de docencia, investigación y continuidad operativa para laboratorios de telemática y redes de computadores.

Palabras clave: laboratorio remoto; redes de datos; telemática; control web; Node.js; virtualización; reserva de dispositivos; educación en ingeniería.

4.1. Introducción

La configuración de una red de datos exige interacción con dispositivos, terminales, topologías y secuencias reales de operación. Un estudiante puede memorizar protocolos, puertos, tablas de encaminamiento o conceptos de VLAN; sin embargo, la comprensión cambia cuando debe enfrentarse a un dispositivo concreto, a una topología específica y a decisiones de configuración con efectos observables.

El problema aparece cuando el laboratorio físico deja de estar disponible o cuando el acceso se vuelve limitado por horario, aforo, distancia o contingencias institucionales. En ese escenario, la docencia en redes puede reducirse a simulación o a teoría sin contacto con el sistema real. El desarrollo documentado en este capítulo responde a esa situación. Se trata de una plataforma diseñada para controlar dispositivos de red de forma remota desde un navegador, manteniendo una estructura de laboratorio con roles diferenciados, reservas, prácticas programadas y acceso restringido [1, 2, 3, 4].

Su relevancia no es únicamente funcional. El sistema fue diseñado como software abierto y no propietario, orientado a formación e investigación, con posibilidad de modificación futura según necesidades pedagógicas y técnicas [1]. Esta decisión instala una capacidad institucional para experimentar, adaptar y ampliar laboratorios remotos.

La documentación técnica sitúa el origen inmediato del problema en la pandemia, cuando la virtualización de la docencia impidió el acceso presencial al laboratorio de telemática y obligó a buscar una alternativa que no se limitara a simuladores, sino que permitiera el control efectivo de dispositivos reales y virtuales [4]. El sistema surge así como respuesta a una necesidad académica concreta.

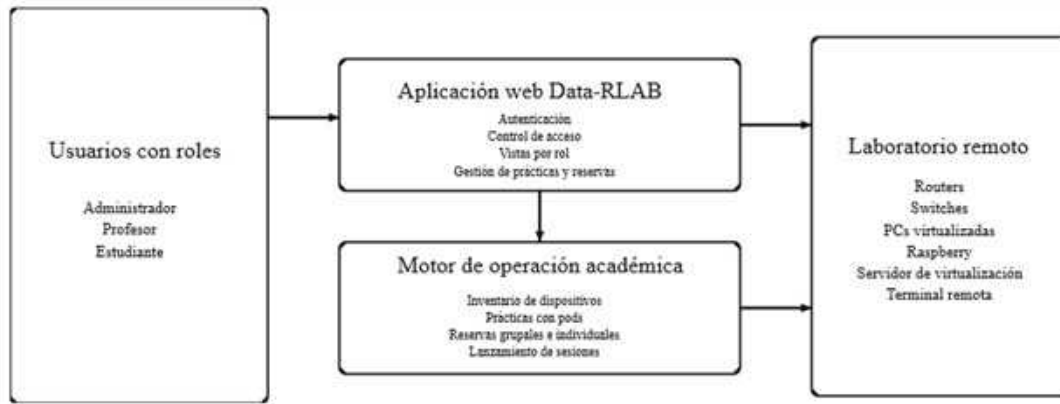


Figura 4.1: Ecosistema funcional del laboratorio remoto, incluyendo usuarios con roles diferenciados, plataforma web, lógica académica de reservas e infraestructura física y virtual del laboratorio.

4.2. Planteamiento del problema

El problema central puede expresarse así: dado un conjunto de dispositivos de red pertenecientes a un laboratorio académico, se requiere una plataforma que permita a usuarios con diferentes roles gestionar, reservar y controlar remotamente dichos dispositivos desde internet, preservando reglas de acceso, disponibilidad temporal y organización pedagógica de prácticas.

Si se define el conjunto de usuarios como

$$\mathcal{U} = \{u_1, u_2, \dots, u_N\},$$

y el conjunto de dispositivos disponibles como

$$\mathcal{D} = \{d_1, d_2, \dots, d_M\},$$

entonces el sistema debe implementar una función de acceso controlado

$$A : \mathcal{U} \rightarrow \mathcal{R},$$

donde $\mathcal{R} = \{\text{administrador, profesor, estudiante}\}$ es el conjunto de roles permitidos. A partir de ese rol, cada usuario dispone de un subconjunto específico de acciones.

A nivel operativo, una práctica puede modelarse como una entidad

$$p = (\mathcal{D}_p, \tau, \rho),$$

donde $\mathcal{D}_p \subseteq \mathcal{D}$ es el subconjunto de dispositivos requeridos, τ representa su ventana temporal de ejecución y ρ indica si la práctica se organiza en pods o como uso unitario de recursos. El sistema debe garantizar que dos reservas incompatibles no usen simultáneamente el mismo recurso, es decir, que para cualesquiera dos reservas r_i y r_j ,

$$\mathcal{D}_{r_i} \cap \mathcal{D}_{r_j} \neq \emptyset \implies \tau_i \cap \tau_j = \emptyset.$$

El problema no consiste solo en mostrar una interfaz para acceder a dispositivos. Consiste en construir una capa de software que traduzca un laboratorio físico a una infraestructura académica accesible remotamente, donde existan inventario, autenticación, reglas de uso, trazabilidad de prácticas y mecanismos reales de control sobre la red.

4.3. Sentido pedagógico del desarrollo

La solicitud institucional del software indica que la novedad del sistema consiste en permitir que estudiantes e investigadores controlen dispositivos de red de forma remota, reproduciendo las características esenciales de un laboratorio presencial, pero con menor exposición a riesgos sobre acceso, software y hardware durante la experimentación [1]. El desarrollo no reemplaza el laboratorio; traslada sus funciones críticas a una plataforma gestionable por internet.

Desde el punto de vista pedagógico, esto habilita al menos tres dimensiones. La primera es continuidad: el estudiante puede seguir trabajando con topologías y configuraciones aun cuando el acceso físico no sea viable. La segunda es seguridad: la práctica ocurre en un entorno mediado por reglas de reserva y control. La tercera es flexibilidad: el sistema permite redefinir dispositivos, prácticas, horarios y combinaciones según necesidades del proceso educativo [1, 3].

En esta lógica, la plataforma funciona como una interfaz entre la enseñanza y la infraestructura. El laboratorio puede entenderse así como un sistema de asignación, autenticación, disponibilidad y control remoto, además de un espacio físico con equipos.

4.4. Arquitectura funcional del sistema

La documentación técnica indica que el software fue creado de manera modular y se compone de dos grandes bloques: una interfaz gráfica de alto nivel y un código de servidor encargado de la lógica de operación, siendo el frontend desarrollado en HTML5 y el backend en Node.js [4]. Sobre esa base, la arquitectura funcional puede representarse como

$$\mathcal{S} = \mathcal{F} \circ \mathcal{B} \circ \mathcal{P},$$

donde \mathcal{F} representa la capa de interacción web, \mathcal{B} la lógica del servidor y \mathcal{P} la infraestructura del laboratorio.

De forma más detallada, puede proponerse la tupla

$$\mathcal{A} = (\mathcal{U}, \mathcal{F}, \mathcal{B}, \mathcal{DB}, \mathcal{I}),$$

donde:

\mathcal{U} = usuarios y roles,

\mathcal{F} = frontend web e interfaz de vistas,

\mathcal{B} = backend de negocio, rutas y control de sesiones,

\mathcal{DB} = persistencia en base de datos,

\mathcal{I} = integración con laboratorio y sistemas externos.

El flujo básico de operación puede modelarse así. Si un usuario autenticado u solicita una acción a , el sistema evalúa primero la autorización según su rol:

$$\alpha(u, a) = \begin{cases} 1, & \text{si el rol de } u \text{ permite } a, \\ 0, & \text{en otro caso.} \end{cases}$$

Solo si $\alpha(u, a) = 1$, la acción se envía a la lógica de negocio, se valida su consistencia temporal y se propaga hacia el conjunto de dispositivos o servicios requeridos.

La interfaz visible es, por tanto, solo la capa externa de una estructura que integra autorización, reserva, persistencia e interacción con equipos y servicios.

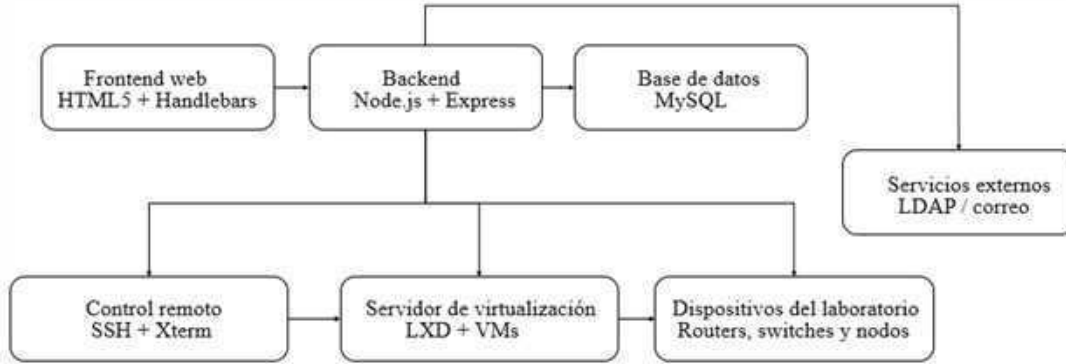


Figura 4.2: Arquitectura lógica del laboratorio remoto, incluyendo frontend, backend, persistencia, autenticación externa y mecanismos de control remoto sobre la infraestructura física y virtual.

4.5. Roles, prácticas y lógica de operación

El manual de usuario organiza la experiencia del sistema según tres actores: administrador, profesor y estudiante. El administrador puede agregar, editar y eliminar dispositivos; el profesor puede crear prácticas, reservarlas e iniciarlas; y los tres roles pueden consultar prácticas reservadas, gestionar reservas individuales e iniciar sesiones correspondientes [3]. Esta organización por roles reproduce la estructura real del laboratorio.

Si se define el conjunto de acciones por rol como

$$\mathcal{A}_{adm}, \quad \mathcal{A}_{prof}, \quad \mathcal{A}_{est},$$

el sistema satisface la relación de inclusión

$$\mathcal{A}_{est} \subseteq \mathcal{A}_{prof} \cup \mathcal{A}_{adm},$$

pero no la igualdad, porque existen privilegios exclusivos de administración y de gestión docente.

Un concepto central es el de *pod*. El manual lo describe como un conjunto de dispositivos de red requerido para que un grupo de estudiantes desarrolle una práctica, por ejemplo switches, routers y PCs necesarios para un ejercicio sobre VLANs [3]. De esta forma, una práctica no se modela solo como una actividad, sino como una asignación estructurada de recursos replicables.

Si una práctica p requiere un conjunto base \mathcal{D}_p de dispositivos y se desean k pods para k

grupos, la demanda total inducida puede expresarse como

$$\mathcal{D}_p^{tot} = \bigcup_{j=1}^k \mathcal{D}_p^{(j)},$$

donde cada $\mathcal{D}_p^{(j)}$ es una réplica funcional del conjunto de recursos de la práctica para el grupo j . Esta idea convierte al sistema en una herramienta para orquestar recursos de laboratorio según estructuras pedagógicas.

4.6. Diseño de software y modularidad

El manual técnico documenta una implementación basada en Node.js 14.16.1, NPM, Express, Handlebars y MySQL, apoyada además por un conjunto amplio de dependencias, entre ellas `bcryptjs`, `express-session`, `passport`, `node-ssh`, `nodemailer` y `jsonwebtoken` [4]. Esta selección tecnológica muestra tres prioridades: autenticación segura, organización modular del servidor e integración remota con infraestructura externa.

La estructura del proyecto se divide en carpetas como `src`, `public`, `views` y `node_modules`, y dentro de `src` aparecen controladores y rutas separados por tipo de usuario y por funcionalidades comunes [4]. Desde el punto de vista arquitectónico, esto puede verse como una descomposición en capas:

$$\mathcal{B} = \mathcal{R}_{adm} \cup \mathcal{R}_{prof} \cup \mathcal{R}_{est} \cup \mathcal{R}_{auth} \cup \mathcal{R}_{common},$$

donde cada subconjunto de rutas implementa parte de la lógica del sistema.

El manual técnico hace explícitas rutas como:

`/admin/add`, `/practice/add`, `/reserve/create`, `/signin`, `/logout`,

y otras asociadas a edición, borrado, reservas y control de dispositivos [4]. Esta estructura revela un desarrollo organizado como aplicación mantenible.

La modularidad también se evidencia en la separación de vistas. El frontend usa HTML5 y Handlebars para generar páginas por rol y componentes comunes como mensajes y navegación [4]. Esta decisión favorece reutilización y hace más fácil evolucionar la interfaz sin reescribir la lógica del sistema.

4.7. Persistencia e integración con otros sistemas

Los datos del sistema se almacenan en MySQL mediante tablas que representan usuarios, dispositivos, prácticas, cursos, reservas individuales, reservas grupales, dispositivos por práctica y matrículas, entre otras [4]. Esa estructura es importante porque la lógica pedagógica del sistema no está codificada solo en pantallas, sino en relaciones persistentes.

Si se denota por \mathcal{T} el conjunto de tablas principales, entonces

$$\mathcal{T} = \left\{ \begin{array}{l} \text{USER, GENERALDEVICE, SPECIFICDEVICE, PRACTICE,} \\ \text{RESERVEG, RESERVEI, RESERVEBYDEVICE,} \\ \text{COURSE, ENROLLMENT} \end{array} \right\}.$$

Cada reserva, práctica o dispositivo existe así como entidad gestionable, auditable y reasignable.

La integración externa también es relevante. El sistema se conecta con el servidor LDAP del LIS para validar el registro de usuarios y, además, se integra con mecanismos de control por terminal mediante Xterm, con un servidor de virtualización basado en LXD y con una Raspberry conectada a parte de la infraestructura del laboratorio [4]. Esto implica que la aplicación opera como orquestador de un ecosistema técnico más amplio.

En términos formales, esa integración puede verse como una función de enlace

$$\Gamma : \mathcal{S} \rightarrow \{\text{LDAP, Xterm, LXD, Raspberry, servidor}\},$$

de modo que una acción iniciada desde el navegador puede terminar en una sesión de terminal sobre un router, un switch o una máquina virtual.

4.8. Topología y virtualización del laboratorio

El sistema se apoya en una topología de laboratorio concreta. El manual técnico indica que actualmente se puede desarrollar una práctica con dos routers, switches de nivel 2 y PCs, organizada lógicamente para estudiar VLANs [4]. A nivel físico, parte de los dispositivos se conectan a una Raspberry y a un servidor Ubuntu donde también se despliega la aplicación.

Esta doble capa, lógica y física, puede modelarse como

$$\mathcal{L} = (\mathcal{N}, \mathcal{E}),$$

donde \mathcal{N} representa nodos del laboratorio y \mathcal{E} los enlaces entre ellos. En este caso, \mathcal{N} incluye routers, switches, servidor, Raspberry y máquinas virtuales.

La virtualización agrega otra dimensión. El sistema utiliza LXD para crear máquinas virtuales que actúan como PCs del laboratorio y pueden asociarse a VLANs determinadas [4]. Esto permite extender el laboratorio sin depender exclusivamente de hardware físico. Si se define

$$\mathcal{V} = \{v_1, v_2, \dots, v_q\}$$

como el conjunto de máquinas virtuales disponibles, cada práctica puede usar un conjunto mixto

$$\mathcal{D}_p = \mathcal{D}_p^{phys} \cup \mathcal{V}_p.$$

Esta combinación entre equipos físicos y virtuales equilibra realismo, escalabilidad y costo.

4.9. Experiencia de usuario y valor del sistema

El manual de usuario describe una experiencia operativa completa: inicio de sesión, registro, vistas por rol, creación de prácticas, reserva de dispositivos, arranque de sesiones y acceso a terminales remotas [3]. El sistema conduce al usuario desde la identidad hasta el recurso, y desde el recurso hasta la sesión activa.

Ese recorrido puede resumirse mediante la secuencia

$$u \rightarrow \text{autenticación} \rightarrow \text{rol} \rightarrow \text{práctica o reserva} \rightarrow \text{dispositivo} \rightarrow \text{control remoto}.$$

Cada transición reduce la incertidumbre de operación: quién es el usuario, qué puede hacer, qué ha reservado, qué recurso le corresponde y cómo accede a él. Esta lógica es consistente con el uso pedagógico de un laboratorio remoto.

La certificación de uso aporta además un dato importante: el sistema cuenta con licencia de uso para el grupo GEPAR y para la coordinación del laboratorio de Telemática de la Facultad de Ingeniería, lo que confirma que no se trata de un desarrollo nominal, sino de una herramienta institucionalmente reconocida [5].



Figura 4.3: Vista principal de la interfaz y navegación inicial del sistema según roles y operaciones disponibles.

4.10. Discusión

El alcance del desarrollo no se reduce a haber habilitado acceso en línea a un laboratorio. Su contribución principal consiste en haber reconstruido en software la lógica de acceso, organización, reserva y ejecución de prácticas de un laboratorio real. Un simulador reemplaza dispositivos por modelos; aquí, en cambio, el software articula usuarios, recursos, sesiones y topologías reales o virtualizadas.

El momento histórico de su aparición también es relevante. Nace en un contexto de virtualización forzada de la docencia, pero no queda limitado a él. Una vez resuelto el problema inicial, el sistema se convierte en una infraestructura reutilizable para educación híbrida, acceso extendido y experimentación asincrónica. Esto le da un valor estratégico mayor que el de una solución puramente coyuntural.

Otra fortaleza del sistema es su apertura técnica. La solicitud institucional señala que, al no ser propietario y al haber sido concebido con módulos reutilizables, permite nuevas

líneas de desarrollo y adaptación pedagógica [1]. Esa afirmación es consistente con la arquitectura modular descrita en el manual técnico [4]. En consecuencia, el desarrollo no opera como un producto cerrado, sino como una plataforma con capacidad de evolución.

4.11. Conclusiones

El sistema documentado en este capítulo muestra que un laboratorio de redes puede ser reconfigurado como infraestructura digital de acceso remoto sin perder sus componentes esenciales de organización, control y práctica. La plataforma permite gestionar dispositivos, reservar recursos, definir prácticas y lanzar sesiones sobre equipos físicos y virtuales mediante una arquitectura web modular.

El desarrollo combina una necesidad docente concreta con una realización tecnológica madura: autenticación, roles, frontend diferenciado, backend estructurado, persistencia en base de datos, integración con LDAP, control vía terminal y virtualización. Esto le da valor no solo como software, sino como capacidad instalada para la universidad.

Dentro de la lógica editorial del Seminario GeoLimna–GEPAR, este capítulo muestra una forma clara de trabajo ingenieril: una necesidad operativa real traducida en una arquitectura técnica verificable, reutilizable y pedagógicamente útil.

Agradecimientos

Se agradece a la Universidad de Antioquia, al laboratorio de Telemática de la Facultad de Ingeniería y al grupo de investigación GEPAR por el entorno académico y técnico que hizo posible el desarrollo de este sistema. Se reconoce además la participación de los autores consignados en el registro de la DNDA y el acompañamiento institucional que permitió validar su uso en contexto real [2, 5].

Referencias

- [1] D. S. Fernández Mc Cann y F. d. J. Vélez Macías, «Solicitud para asignación de puntos por desarrollo de software: DATA-RLAB, Laboratorio remoto de datos,» Universidad de Antioquia, Medellín, Colombia, Documento institucional, 3 de abr. de 2023.
- [2] G. A. Marín Lopera, D. S. Fernández Mc Cann, J. A. Ríos Pérez, S. López Stan e Y. D. Ballesteros, *DATA-RLAB*, Registro ante la Dirección Nacional de Derecho de Autor, Libro 13, Tomo 89, Partida 261, fecha de registro: 31-mar-2022, 2022.
- [3] Universidad de Antioquia, *Data-RLAB: Manual de usuario*, Universidad de Antioquia, Medellín, Colombia, 2021.
- [4] Universidad de Antioquia, *Data-RLAB: Manual técnico*, Universidad de Antioquia, Medellín, Colombia, 2021.
- [5] G. A. Marín Lopera y D. S. Fernández Mc Cann, «Certificación de uso del software DATA-RLAB,» Universidad de Antioquia, Medellín, Colombia, Certificación institucional, 4 de abr. de 2023.

Capítulo 5

Estimación objetiva de la frecuencia respiratoria en larvas de megalóptero mediante procesamiento digital de imágenes

Autores

Isabella Bermón Rojas¹; Juan Pablo Serna López²; David Stephen Fernández Mc Cann³.

Resumen

La respiración de organismos sensibles a perturbaciones ambientales puede funcionar como biomarcador de estrés y, por tanto, como una vía objetiva para estudiar calidad del agua. En este capítulo se presenta un desarrollo computacional orientado a estimar la frecuencia respiratoria de larvas de megalóptero mediante el análisis de video obtenido bajo estereomicroscopio. El sistema permite al usuario cargar un video, seleccionar una región de interés sobre la zona branquial, procesar cada fotograma, extraer la variación del área asociada al movimiento respiratorio y construir una señal temporal a partir de la cual se obtiene la frecuencia media de respiración. La propuesta se materializó en una interfaz gráfica que integra carga de video, selección de región, ejecución del análisis,

¹Estudiante de Ingeniería Electrónica y Telecomunicaciones.

²Profesor, Departamento de Ingeniería Ambiental y Sanitaria.

³Profesor, Departamento de Ingeniería Electrónica y Telecomunicaciones.

^{1,2,3}Universidad de Antioquia. Grupos de investigación GEPAR y GeoLimna.

visualización de la curva respiratoria y exportación de resultados en formato tabular. El desarrollo responde a un problema metodológico concreto: la medición manual de la respiración en megalópteros es lenta, subjetiva y susceptible a errores inducidos por la frecuencia de obturación de la cámara, la persistencia visual y la dificultad de seguir movimientos finos en organismos vivos. El capítulo reconstruye el problema, formaliza el método, describe la arquitectura del software y discute su valor como herramienta para estudios ecotoxicológicos y de bioindicación.

Palabras clave: megalóptero; bioindicadores; frecuencia respiratoria; procesamiento digital de imágenes; biomarcadores; calidad del agua.

5.1. Introducción

En el estudio de ecosistemas acuáticos, algunos organismos permiten inferir condiciones del medio a partir de su comportamiento, permanencia o respuesta fisiológica. Los megalópteros pertenecen a este grupo de organismos sensibles a perturbaciones ambientales y, por ello, resultan útiles como bioindicadores [1, 2, 3, 4]. Esa utilidad, sin embargo, depende de la posibilidad de medir su respuesta con suficiente objetividad.

Una de esas respuestas es la respiración. El movimiento rítmico de las branquias abdominales expresa una dinámica fisiológica que puede alterarse frente a cambios del ambiente o a la exposición a contaminantes. La cuantificación convencional de esa respiración suele hacerse de manera manual, observando directamente al organismo bajo el microscopio o revisando el video cuadro a cuadro. La documentación del proyecto señala precisamente esta dificultad y advierte que tanto la frecuencia de obturación de la cámara como la capacidad de retención visual del observador pueden introducir errores apreciables [1].

Frente a ese problema, el desarrollo aquí presentado propone una solución concreta: transformar el movimiento branquial en una señal visual cuantificable. Para ello, el software permite seleccionar un segmento del video donde el movimiento de la branquia sea visible, procesa cada fotograma, estima el área ocupada por la estructura respiratoria y almacena esa evolución temporal en un vector, a partir del cual genera una gráfica y calcula una frecuencia media de respiración [1, 2, 3]. El aporte del sistema consiste en formalizar una observación biológica y convertirla en una magnitud reproducible.

El desarrollo cuenta además con una trazabilidad institucional clara. Dispone de registro ante la DNDA, certificación de uso por parte de los grupos GeoLimna y GEPAR, manual de usuario, manual técnico y una trayectoria de circulación académica que incluye una ponencia sobre análisis de frecuencia respiratoria en megalópteros mediante procesamiento digital de imágenes [2, 3, 5, 6, 7]. Esto lo sitúa como un caso de investigación aplicada

convertida en producto tecnológico.



Figura 5.1: Larva de megalóptero empleada como referencia visual del conjunto de trabajo, con énfasis en la región abdominal donde se manifiesta el movimiento respiratorio analizado por el software.

5.2. Problema de medición y formulación general

El problema central puede formularse de la siguiente manera: dado un video de una larva de megalóptero observado mediante estereomicroscopio, se requiere estimar su frecuencia respiratoria a partir del movimiento periódico de una región branquial visible, de forma objetiva, reproducible y menos sensible al error humano que la observación manual.

Sea

$$V = \{I_1, I_2, \dots, I_N\}$$

el conjunto ordenado de fotogramas del video, donde cada I_k representa la imagen correspondiente al instante t_k . El usuario selecciona sobre el primer fotograma una región de interés

$$\Omega \subset \mathbb{R}^2,$$

que corresponde al área espacial donde se observará el movimiento respiratorio. Esta región acota el problema biológico y lo vuelve computable: el software no intenta interpretar toda la larva al mismo tiempo, sino una zona donde la expansión y compresión de la branquia pueda seguirse con estabilidad [2, 3].

Una vez definida la región, cada fotograma se procesa para producir una medida escalar

de área

$$A_k = A(t_k),$$

que representa el área segmentada de la estructura respiratoria en el instante t_k . El conjunto

$$\mathbf{A} = \{A_1, A_2, \dots, A_N\}$$

constituye entonces la señal respiratoria del individuo. Cuando la branquia se expande, el área aumenta; cuando se contrae, el área disminuye. La respiración queda representada como una oscilación medible.

Si se identifican en la señal los máximos respiratorios $\{p_1, p_2, \dots, p_M\}$, la frecuencia media puede aproximarse por

$$f_r = \frac{M}{T},$$

donde M es el número de respiraciones detectadas y T el tiempo total analizado, expresado en minutos o segundos según la convención elegida. Esta ecuación resume el principio del sistema: contar respiraciones no por percepción directa del observador, sino por eventos detectados en una señal derivada de imágenes.

5.3. Fundamento biológico del método

El desarrollo responde a una necesidad dentro del estudio de bioindicadores acuáticos. La documentación del software y el manuscrito asociado destacan que las larvas de megalóptero son organismos sensibles a cambios en las condiciones ambientales y, por ello, útiles para estudios de calidad del agua [2, 3, 4]. En particular, el manuscrito recuerda que en el caso del género *Corydalus*, el movimiento ventilatorio implica retracción y prolongación rítmica de pares de branquias abdominales, lo que justifica centrar la observación sobre regiones branquiales visibles [4].

Esa base biológica da sentido a la representación matemática anterior. La variable $A(t)$ no se interpreta como un área abstracta, sino como una proyección visual del movimiento respiratorio. Del mismo modo, los máximos de la señal no son picos numéricos arbitrarios, sino eventos fisiológicos asociados a ciclos de expansión. La validez del algoritmo depende precisamente de ese vínculo entre anatomía, imagen y señal temporal.

5.4. Arquitectura general del método

La lógica del sistema puede representarse como una cadena de cinco etapas:

$$V \xrightarrow{\mathcal{S}} \Omega \xrightarrow{\mathcal{P}} \mathbf{A} \xrightarrow{\mathcal{D}} \mathcal{P}_{\text{máx}} \xrightarrow{\mathcal{E}} f_r,$$

donde \mathcal{S} representa la selección de la región de interés, \mathcal{P} el procesamiento de cada fotograma, \mathcal{D} la detección de máximos sobre la señal de área y \mathcal{E} la estimación final de la frecuencia respiratoria.

En términos operativos, el método sigue esta secuencia: primero se aísla una porción del organismo donde el movimiento respiratorio sea visible; luego se transforma cada imagen en una representación más limpia del objeto; después se calcula el área segmentada fotograma a fotograma; posteriormente se observan los cambios de esa magnitud en el tiempo; y finalmente se identifican los eventos periódicos que corresponden a respiraciones completas.

La presentación del flujo en la ponencia resume esta lógica mediante una cadena operativa de video a fotogramas, uso de la componente S del espacio HSV, binarización, delimitación de zona de interés, cálculo de variación y visualización del vector de datos [8]. Esta representación muestra que la solución se basa en una secuencia explícita de transformaciones visuales y no en un procedimiento opaco.

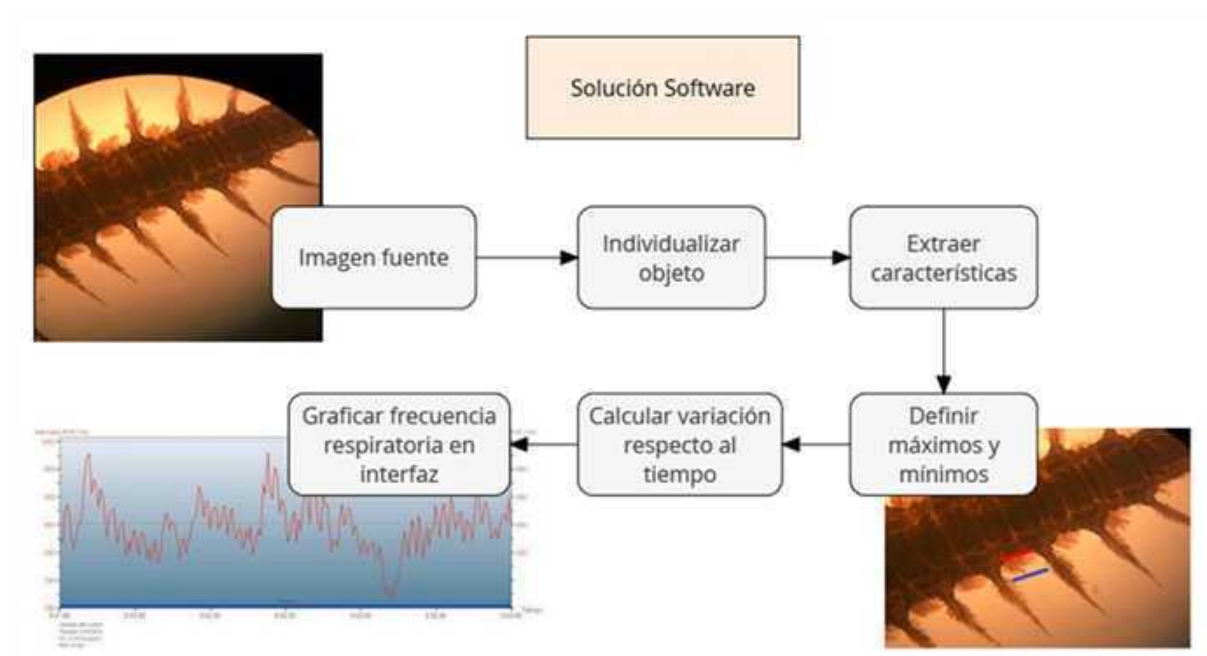


Figura 5.2: Flujo general del método de análisis, desde el video de entrada hasta la construcción de la señal y la estimación de la frecuencia respiratoria.

5.5. Preprocesamiento y segmentación de la región respiratoria

Según la guía de usuario y el manual técnico, el análisis comienza cuando el usuario carga un video y selecciona una región rectangular sobre el primer fotograma [2, 3]. Esa decisión define el dominio espacial de trabajo para todo el video. En términos formales, si $I_k(x, y)$ es el fotograma k , el sistema trabaja con

$$I_k^\Omega(x, y) = I_k(x, y)|_{(x,y) \in \Omega}.$$

A partir de allí, el manuscrito y los materiales de ponencia muestran que se emplean operaciones de segmentación y morfología matemática para extraer el cuerpo del organismo y reducir el ruido del fondo [4]. La propuesta menciona explícitamente el uso de umbralización en espacios de color, emborrado tipo *disk*, binarización y *regionprops*. La guía de usuario añade que el sistema aplica técnicas de teoría de imagen como elemento estructurante, dilatación y erosión para normalizar el fondo y dejar más limpia la región útil [2].

Si se denota por $B_k(x, y)$ la imagen binaria resultante en el fotograma k , una forma compacta de describir esta etapa es

$$B_k = \mathcal{M}(\mathcal{T}(I_k^\Omega)),$$

donde \mathcal{T} simboliza la transformación de color y umbralización, y \mathcal{M} el conjunto de operaciones morfológicas de limpieza. El propósito de esta etapa es asegurar que la medida de área posterior responda principalmente al movimiento respiratorio y no a artefactos del fondo, sombras o ruido del microscopio.

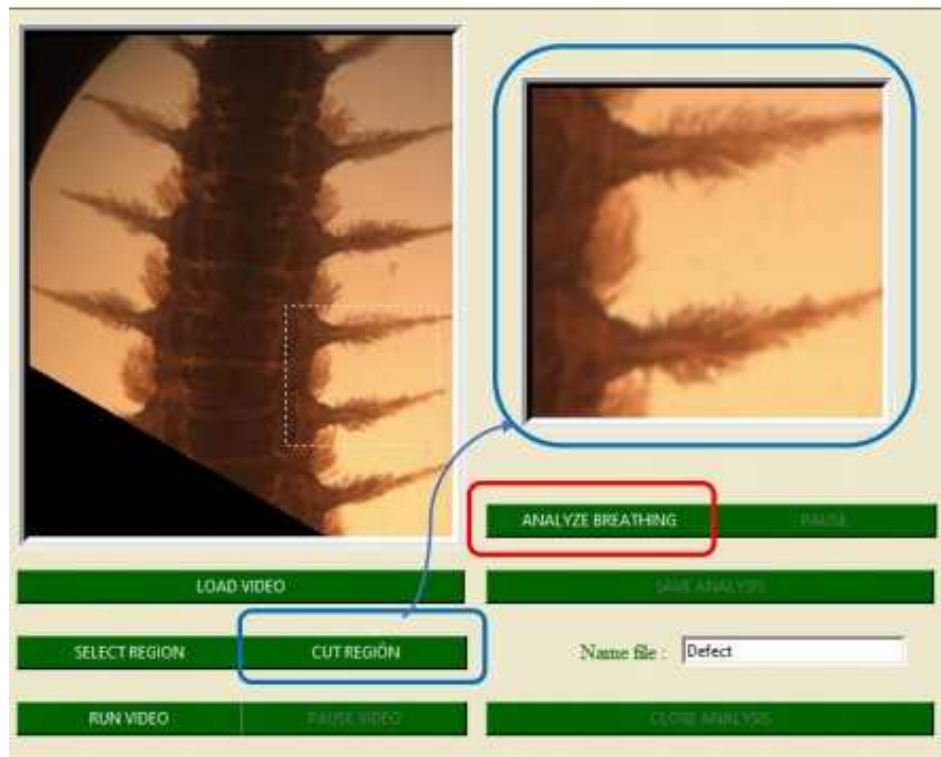


Figura 5.3: Selección de la región de interés sobre la zona respiratoria que será analizada a lo largo del video.

5.6. Construcción de la señal respiratoria

Una vez segmentada la región de interés en cada fotograma, el software calcula una medida de área y la guarda en un vector [1, 2, 3, 5]. De este modo, la respiración se rastrea mediante una magnitud geométrica.

Si $B_k(x, y) \in \{0, 1\}$ es la imagen binaria del fotograma k , el área segmentada puede

expresarse como

$$A_k = \sum_{(x,y) \in \Omega} B_k(x,y).$$

Esta suma cuenta cuántos píxeles activos pertenecen a la estructura observada en ese instante. Cuando la región branquial se expande, A_k tiende a crecer; cuando se contrae, A_k disminuye. La curva $A(t)$ se convierte así en una representación indirecta del patrón respiratorio.

La guía de usuario muestra que durante la ejecución el sistema construye en tiempo real una gráfica de área versus tiempo y, junto a ella, una estimación de la frecuencia respiratoria media [2]. Esa visualización permite verificar que el algoritmo está siguiendo una dinámica coherente y, al mismo tiempo, traduce el procesamiento interno a una forma interpretable por el usuario investigador.

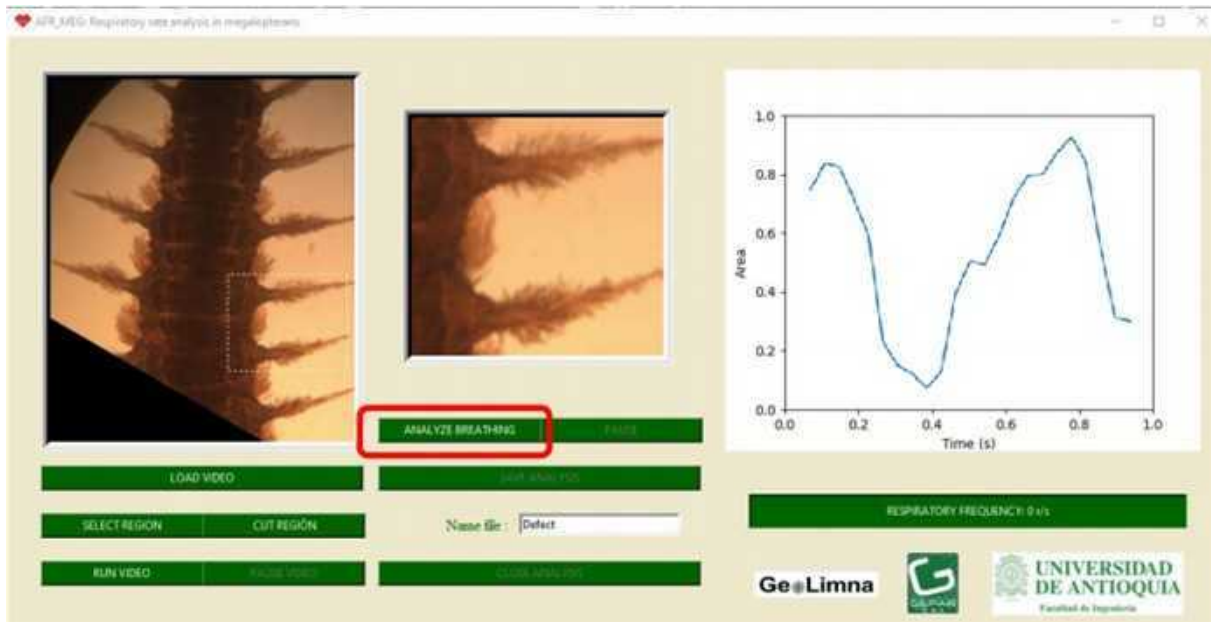


Figura 5.4: Interfaz principal del software durante la ejecución, con video de entrada, región de análisis y gráfico de resultados.

5.7. Detección de respiraciones y estimación de frecuencia

La señal de área debe analizarse para identificar cuántos eventos respiratorios contiene. El manual técnico indica que el sistema utiliza herramientas como `find_peaks` dentro de su arquitectura y señala que el vector resultante se analiza para encontrar los picos de

mayor área, asociados a respiraciones completas [3].

Si se define el conjunto de índices de máximos relevantes como

$$\mathcal{P}_{\text{máx}} = \{k_1, k_2, \dots, k_M\},$$

entonces cada k_i representa un instante en el que la señal alcanza un máximo asociado a una expansión respiratoria. Bajo esta representación, la frecuencia media se estima como

$$f_r = \frac{M}{t_{k_M} - t_{k_1}},$$

si se usa el intervalo cubierto por los máximos extremos, o como

$$f_r = \frac{M}{T},$$

si se usa el tiempo total analizado del video. En ambos casos la interpretación es la misma: la frecuencia respiratoria se obtiene dividiendo el número de eventos detectados entre el tiempo de observación.

La automatización reemplaza así una tarea humana fatigante y propensa a error por una detección basada en regularidades de señal. Esto no elimina la necesidad de criterio biológico, pero sí reduce la subjetividad de la medición.

5.8. Interfaz gráfica y experiencia de uso

Uno de los aciertos del desarrollo es haber convertido esta cadena metodológica en una interfaz operable. La guía de usuario describe una ventana principal organizada en tres columnas: a la izquierda se muestra el primer fotograma del video y se gestionan las acciones de carga, selección y reproducción; en el centro aparece la región recortada y los botones de análisis y guardado; y a la derecha se visualiza la gráfica de área contra tiempo y el valor medio de respiraciones [2].

Esa organización espacial refleja la secuencia lógica del método: observar, seleccionar, procesar y leer resultados. El usuario no se enfrenta a una caja cerrada, sino a una interfaz que le permite seguir el tránsito desde la imagen cruda hasta la señal cuantificada. Además, el sistema exporta resultados en un archivo `.csv`, lo que facilita tratamientos estadísticos posteriores [2, 3].

5.9. Arquitectura de software

El manual técnico señala que el sistema fue desarrollado con Python 3.7.3, PyQt5 y OpenCV, y describe una estructura modular compuesta por bibliotecas, una clase principal `MainWindow` y un conjunto de funciones externas para procesamiento de imágenes [3]. Aunque el documento mezcla terminología entre PyQt y Tkinter, su sentido general es claro: la aplicación se construyó como una interfaz gráfica que orquesta el flujo completo de adquisición, procesamiento, visualización y almacenamiento.

Una forma compacta de representar la arquitectura es mediante la tupla

$$\mathcal{A} = (\mathcal{G}, \mathcal{V}, \mathcal{P}, \mathcal{R}),$$

donde \mathcal{G} es la capa gráfica, \mathcal{V} el módulo de manejo de video, \mathcal{P} el conjunto de rutinas de procesamiento digital de imágenes y \mathcal{R} el módulo de resultados y exportación. Esta representación resume lo que hace el sistema: recibir un video, transformarlo en una señal y entregar una salida legible y exportable.

El manual técnico documenta además funciones como `callback()`, `update_image()`, `recortar()`, `create_database()` y `Control()`, siendo esta última la que concentra el preprocesamiento de cada fotograma, la segmentación, el emparejamiento de plantillas, la detección de contornos y el cálculo del cambio de área [3]. Esta modularidad muestra que el desarrollo se materializó como una aplicación mantenible y ampliable.

5.10. Alcance científico y valor del desarrollo

El formulario de solicitud institucional enfatiza que antes del software no se contaba con una herramienta que permitiera medir objetivamente la frecuencia respiratoria de un megalóptero, y que la práctica habitual seguía siendo manual [1]. Esta afirmación da al desarrollo una dimensión metodológica clara: no solo produce una herramienta informática, sino que introduce una forma cuantitativa de medir una respuesta fisiológica.

Además, su aplicación potencial trasciende el caso puntual de una larva. La solicitud destaca que la frecuencia respiratoria del megalóptero puede utilizarse en estudios biológicos para determinar calidad del agua y para evaluar impacto de contaminantes en acuíferos [1]. La variable fisiológica estimada por el software se convierte así en una entrada para preguntas ambientales de mayor alcance.

También es relevante que el sistema haya sido presentado en escenarios académicos especializados, como la ponencia sobre análisis de frecuencia respiratoria de un megalóptero

usando procesamiento digital de imágenes [7]. Ese recorrido refuerza la condición del desarrollo como contribución con proyección científica y técnica.

Time	Area	Average respiration
0	2979	2
0.03348018	3049	2
0.06696035	2993	2
0.10044053	2961.5	2
0.1339207	2882	2
0.16740088	2504	2
0.20088106	2424	2
0.23436123	2391.5	2
0.26784141	2369	2
0.30132159	2406.5	2
0.33480176	2578.5	2
0.36828194	2644.5	2
0.40176211	2651	2
0.43524229	2703.5	2
0.46872247	2808.5	2
0.50220264	2880.5	2
0.53568282	2940.5	2
0.569163	3007	2
0.60264317	3074.5	2
0.63612335	3041	2
0.66960352	2705.5	2

Figura 5.5: Archivo CSV exportado por el sistema con la tabla de tiempo, área y respiración media, como evidencia de resultados reutilizables.

5.11. Discusión

El principal mérito del desarrollo consiste en haber conectado tres dominios que suelen aparecer separados: la bioindicación, la visión por computador y la instrumentación metodológica. El problema biológico es la respiración de un organismo sensible a la contaminación. El problema ingenieril es convertir ese movimiento en datos. El software opera precisamente como interfaz entre ambos.

Otra fortaleza importante es la elección de una representación interpretable. En lugar de inferir la frecuencia respiratoria con un modelo opaco, el sistema construye una señal geométrica de área y busca eventos sobre ella. Esta decisión aporta trazabilidad. El investigador puede ver la región seleccionada, la curva resultante y el archivo exportado, y

puede juzgar si la medición es coherente.

El desarrollo no elimina todos los retos. La estabilidad del video, la calidad del enfoque, la correcta elección de la región y el comportamiento del organismo influyen en el resultado. Precisamente por eso el sistema tiene valor: no oculta la complejidad, sino que la organiza en un flujo operativo más robusto que la observación manual.

5.12. Conclusiones

Este capítulo presentó un desarrollo orientado a la estimación objetiva de la frecuencia respiratoria en larvas de megalóptero mediante procesamiento digital de imágenes. El sistema transforma una observación biológica compleja en una señal temporal derivada del área segmentada de una región branquial, y a partir de ella calcula una frecuencia media de respiración.

La propuesta combina selección de región de interés, segmentación por color y morfología, construcción de una señal de área, detección de máximos y exportación de resultados. En consecuencia, el desarrollo no se reduce a una interfaz gráfica, sino que integra una metodología cuantitativa encapsulada en software.

Desde la perspectiva del Seminario GeoLimna–GEPAR, este capítulo muestra con claridad una línea de trabajo en la que un problema real del entorno ambiental y biológico se traduce en un sistema medible, verificable y útil. El desarrollo automatiza una tarea y, al mismo tiempo, instala una capacidad de observación para la universidad en biomarcadores visuales y analítica ecológica asistida por imagen.

Agradecimientos

Se agradece a la Universidad de Antioquia y a los grupos de investigación GeoLimna y GEPAR por el entorno académico y técnico que hizo posible este desarrollo. Se reconoce además la participación de los autores registrados en la DNDA y el uso institucional certificado del software por parte de ambos grupos [5, 6].

Referencias

- [1] D. S. Fernández Mc Cann y F. d. J. Vélez Macías, «Solicitud para asignación de puntos por desarrollo de software: MEGAREF, Estimador de frecuencia respiratoria en megalópteros,» Universidad de Antioquia, Medellín, Colombia, Documento institucional, 3 de abr. de 2023.
- [2] Universidad de Antioquia, *MegaREF: Megaloptera Respiratory Frequency Analysis. User's Guide*, Universidad de Antioquia, Medellín, Colombia, 2021.
- [3] Universidad de Antioquia, *MegaREF: Megaloptera Respiratory Frequency Analysis. Service Manual*, Universidad de Antioquia, Medellín, Colombia, 2021.
- [4] J. P. Serna López, I. Bermón Rojas y D. S. Fernández Mc Cann, «Software para el análisis de la frecuencia respiratoria de un megalóptero usando el procesamiento digital de imágenes,» Manuscrito en desarrollo, 2022.
- [5] D. S. Fernández Mc Cann, J. P. Serna Lopez e I. Bermón Rojas, *MEGAREF: Megaloptera Respiratory Frequency Analysis*, Registro ante la Dirección Nacional de Derecho de Autor, Libro 13, Tomo 90, Partida 96, fecha de registro: 01-jun-2022, 2022.
- [6] F. d. J. Vélez Macías y D. S. Fernández Mc Cann, «Certificación de uso del software MEGAREF,» Universidad de Antioquia, Medellín, Colombia, Certificación institucional, 4 de abr. de 2023.
- [7] I. Bermón, J. P. Serna López y D. Fernández, *Software para el análisis de la frecuencia respiratoria de un megalóptero usando el procesamiento digital de imágenes*, Ponencia presentada en el V Congreso Macrolatinos, Panamá, 22 de octubre de 2021, 2021.
- [8] *Flujo metodológico MegaREF*, Gráficos de apoyo del proyecto MegaREF, 2022.

Capítulo 6

Generación sintética y realista de matrículas vehiculares para entrenamiento de algoritmos de reconocimiento de patrones

Autores

Carlos Alfonso Arbeláez Acevedo¹; Gustavo Andrés Marín Lopera²; David Stephen Fernández Mc Cann³.

Resumen

Los algoritmos de reconocimiento de matrículas vehiculares dependen de la disponibilidad de datos diversos, abundantes y visualmente plausibles. Sin embargo, construir bases de datos reales suficientemente grandes suele implicar costos de adquisición, problemas de privacidad, sesgos de captura y una variabilidad difícil de controlar experimentalmente. En este capítulo se presenta un desarrollo orientado a resolver ese cuello de botella: un generador de matrículas que parte de placas sintéticas ideales y produce imágenes con apariencia realista, alterando además atributos como brillo, desenfoque, color, ciudad y texto de la placa. La herramienta fue concebida para la creación masiva de datasets y

¹Estudiante, Departamento de Ingeniería Electrónica y Telecomunicaciones.

²Profesor y coordinador del laboratorio de Telemática.

³Profesor, Departamento de Ingeniería Electrónica y Telecomunicaciones.

^{1,2,3}Universidad de Antioquia. Grupo de investigación GEPAR.

para el entrenamiento de algoritmos de visión por computador, aprendizaje automático y redes neuronales convolucionales. El sistema integra un generador sintético de placas, una interfaz gráfica de manipulación y una arquitectura generativa basada en traducción imagen a imagen mediante CycleGAN. Además de su uso en investigación, el desarrollo ha resultado útil en docencia, pues permite construir ejemplos controlados, demostrar efectos de variación visual sobre el reconocimiento automático y formar a los estudiantes en el diseño de datasets para aprendizaje supervisado. El capítulo describe el problema, formaliza la lógica del generador, presenta la arquitectura del software y discute su alcance como herramienta de investigación, prototipado y enseñanza.

Palabras clave: placas vehiculares; generación sintética; datasets; CycleGAN; redes convolucionales; reconocimiento de patrones; visión por computador; docencia en inteligencia artificial.

6.1. Introducción

En visión por computador, muchos modelos no fallan por falta de diseño matemático, sino por falta de datos adecuados. Una arquitectura convolucional puede ser correcta y un procedimiento de entrenamiento puede estar bien ajustado; sin embargo, si el conjunto de imágenes no expresa la diversidad visual del problema real, el desempeño termina siendo frágil. El caso de las matrículas vehiculares es especialmente ilustrativo: variaciones de brillo, desenfoque, suciedad, color, ángulo, desgaste o tipografía afectan de manera directa la capacidad de segmentar, detectar y reconocer.

El desarrollo presentado en este capítulo parte de esa dificultad. La documentación institucional del software indica que LPGe fue creado para generar imágenes de matrículas realistas a partir de placas sintéticas, con atributos variables como número de placa, ciudad, brillo, desenfoque y color, y con una función explícita de generación aleatoria individual o masiva para la construcción de bases de datos [1, 2, 3]. El objetivo no es simplemente dibujar placas, sino construir condiciones de entrenamiento.

Este punto resulta importante tanto en investigación como en docencia. En investigación, porque disponer de un generador parametrizable permite construir conjuntos de entrenamiento, validación y prueba con diferentes grados de dificultad y variación. En docencia, porque permite mostrar que un modelo no aprende solo de una arquitectura, sino también del diseño del dataset. En otras palabras, el sistema no solo produce imágenes; también permite analizar cómo se construye evidencia visual para entrenar sistemas inteligentes.

El desarrollo se apoya además en una base algorítmica actual. El manual de usuario y el manual técnico indican que el sistema está basado en una red generativa adversarial, espe-

cíficamente en la arquitectura CycleGAN para traducción imagen a imagen, entrenada con dos datasets de 580 imágenes, uno sintético y uno real [3, 4, 5]. Esta decisión tecnológica le da al software un doble valor: por una parte, genera salidas plausibles para aplicaciones prácticas; por otra, constituye un caso útil para explicar cómo una arquitectura generativa puede emplearse en transferencia de estilo con propósito ingenieril.



Figura 6.1: Interfaz general del generador, con construcción de la placa sintética a la izquierda y visualización de la versión realista generada a la derecha.

6.2. Problema y motivación del desarrollo

El problema que aborda este desarrollo puede formularse así: dado un sistema de reconocimiento automático de matrículas, se requiere disponer de un conjunto de imágenes suficientemente amplio, variado y realista para entrenar modelos de aprendizaje automático y redes neuronales sin depender exclusivamente de capturas reales.

Si se representa el conjunto de imágenes de entrenamiento como

$$\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N,$$

donde x_i es una imagen de placa e y_i la etiqueta asociada, el reto no es solo aumentar N , sino lograr que el conjunto cubra una diversidad visual suficientemente rica. Una base de datos pobre en variación puede inducir un modelo aparentemente preciso en laboratorio, pero débil frente a escenarios reales.

El desarrollo institucional lo expresa con claridad: la falta de datasets de placas dificulta

entrenar modelos de *machine learning* con miles de imágenes, y construir manualmente una base de datos significativa resulta agotador y costoso [1, 4]. A partir de ahí, el sistema propone una solución basada en generación controlada. La idea no es reemplazar toda imagen real, sino ampliar el espacio de ejemplos posibles mediante una función generativa

$$G : \mathcal{S} \rightarrow \mathcal{R},$$

donde \mathcal{S} es el conjunto de placas sintéticas idealizadas y \mathcal{R} el conjunto de placas con apariencia realista.

Esta formulación resume la lógica del sistema. La entrada sintética ofrece control sobre el contenido semántico de la placa: letras, números, ciudad y color. La salida realista introduce textura, degradación, iluminación y otras irregularidades visuales que aproximan el resultado al entorno operativo. En consecuencia, el sistema no solo produce imágenes; construye un puente entre idealización y realismo.

6.3. Valor didáctico y científico del generador

Aunque el desarrollo fue pensado para la creación de datasets, su utilidad trasciende ese objetivo inicial. En el trabajo con estudiantes, una dificultad frecuente consiste en enseñar que el desempeño de una CNN o de un clasificador clásico no depende únicamente de la arquitectura del modelo, sino también del diseño de los datos. El sistema resuelve parte de ese problema pedagógico porque permite variar controladamente ciertos atributos y observar cómo cambian los resultados del entrenamiento.

Por ejemplo, si se define una familia de datasets

$$\mathcal{D}(\beta, \sigma, c),$$

donde β representa un parámetro de brillo, σ un parámetro de desenfoque y c un atributo de color, el docente puede construir varios escenarios de entrenamiento con exactamente la misma semántica textual, pero con dificultades visuales progresivas. Esto resulta útil para mostrar en clase fenómenos como sobreajuste, sensibilidad a la calidad de imagen, robustez frente a variaciones y transferencia de dominio.

Desde el punto de vista científico, el desarrollo también abre líneas de trabajo. La solicitud institucional indica que el grupo quedó con capacidad para ofrecer servicios de creación de algoritmos de identificación de placas y de entrenamiento de sistemas de IA para dichos reconocimientos [1]. Esta afirmación sitúa al software como una plataforma reutilizable para futuros trabajos en detección, segmentación, OCR, aprendizaje profundo y análisis

comparativo entre modelos.

6.4. Arquitectura conceptual del sistema

La lógica del generador puede describirse como una secuencia de tres etapas: construcción sintética, traducción a realismo y alteración controlada de atributos. Formalmente, puede escribirse como

$$x_{real} = \mathcal{A}(\mathcal{G}(x_{syn})),$$

donde x_{syn} es la placa sintética inicial, \mathcal{G} es el generador basado en traducción imagen a imagen y \mathcal{A} representa un conjunto de alteraciones adicionales como brillo, desenfoque o cambio de color.

La primera etapa produce una placa limpia y controlada, donde el texto y la ciudad son exactamente los que el usuario desea. La segunda transforma esa imagen ideal hacia una apariencia más cercana a una placa fotografiada en condiciones reales. La tercera permite perturbar visualmente la salida para ampliar aún más la variabilidad del dataset. La combinación de estas etapas convierte al software en un generador de ejemplos de entrenamiento y no solo en un editor visual.

En el manual técnico aparece una estructura modular consistente con esta interpretación. El software se divide en tres secciones principales: la arquitectura del modelo generador, el generador sintético de placas y la interfaz desarrollada con PySimpleGUI [4]. Esta separación refleja una buena práctica de diseño: contenido, modelo e interacción quedan desacoplados.

6.5. Generación sintética de la placa

La primera pieza del sistema consiste en producir una representación limpia de la matrícula a partir de parámetros definidos. El manual de usuario indica que el software recibe como entrada el texto de la placa y la ciudad asociada, y construye una versión sintética que luego será transformada [3]. Desde la perspectiva funcional, esto puede expresarse mediante

$$x_{syn} = P(t, m),$$

donde t es la cadena alfanumérica de la placa y m la ciudad o metadato textual mostrado en la parte inferior.

Esta etapa cumple un papel esencial: fija la semántica de la imagen. A diferencia de un

generador completamente libre, el sistema no inventa arbitrariamente el contenido textual; lo conserva. Esto es fundamental para aplicaciones supervisadas, porque permite asociar a cada imagen una etiqueta exacta y consistente. Desde la docencia, además, permite fabricar ejemplos donde solo cambia la apariencia visual, mientras el contenido semántico permanece constante.



Figura 6.2: Ejemplo de generación específica de una placa, mostrando la versión sintética y su correspondiente versión alterada con apariencia realista.

6.6. Traducción de sintético a realista mediante modelo generativo

El núcleo algorítmico del desarrollo está en la etapa de traducción de imagen a imagen. Tanto el manual de usuario como el manual técnico indican que el software se basa en una red generativa adversarial, específicamente CycleGAN, y que fue entrenado con dos datasets de 580 imágenes, uno sintético y uno real [3, 4, 5].

Una forma de representar esta parte del sistema es considerar dos dominios de imágenes:

$$X = \text{dominio de placas sintéticas}, \quad Y = \text{dominio de placas reales}.$$

La tarea del generador consiste en aprender una función

$$G : X \rightarrow Y,$$

capaz de mapear una imagen sintética a una imagen visualmente compatible con el dominio real. En el contexto de CycleGAN, ese aprendizaje no se limita a una transformación directa; se apoya además en una consistencia cíclica entre dominios, lo que puede resumirse

conceptualmente como

$$F(G(x)) \approx x,$$

donde F es la función inversa aproximada de regreso al dominio sintético. La utilidad de esta idea es que la red aprende a transformar estilo y textura sin perder el contenido principal de la placa.

En términos prácticos, esto significa que una placa dibujada de forma limpia puede adquirir manchas, desgaste aparente, diferencias de iluminación y pequeños defectos locales, manteniendo al mismo tiempo la estructura semántica que la vuelve utilizable para entrenamiento. Esa propiedad es la que hace al desarrollo especialmente útil para generación de datasets.

6.7. Alteraciones controladas como expansión del espacio de entrenamiento

El software no se limita a producir una sola imagen realista. El manual de usuario documenta que la interfaz permite modificar brillo, desenfoque y color, incluyendo el paso de blanco a amarillo en la placa [3]. Desde el punto de vista metodológico, esta capacidad convierte al generador en una herramienta para explorar el espacio de variación visual.

Si se denota por x_{gen} la imagen generada por el modelo, una alteración paramétrica puede expresarse como

$$x' = \Phi(x_{gen}; \beta, \sigma, \kappa),$$

donde β representa brillo, σ un nivel de desenfoque y κ una condición cromática. Esta expresión formaliza la idea de que el usuario puede recorrer una familia de imágenes relacionadas mediante parámetros de apariencia.

Esa familia es útil por dos razones. Primero, permite crear datasets más robustos para modelos de reconocimiento automático. Segundo, hace visible, en escenarios de clase, que la generalización visual no es automática. Un algoritmo entrenado solo con imágenes limpias puede degradarse fuertemente frente a placas con desenfoque o saturación. El sistema permite construir demostraciones concretas de ese fenómeno sin depender de campañas complejas de captura.



Figura 6.3: Modificadores visuales de brillo, desenfoque y color aplicados sobre placas generadas para expandir la variabilidad del dataset.

6.8. Generación masiva y construcción de datasets

La potencia práctica del sistema aparece cuando se pasa de la generación individual a la generación masiva. El manual de usuario explica que el software permite definir un número de placas aleatorias, seleccionar una carpeta de salida y lanzar un proceso de generación automática que guarda todas las imágenes resultantes en el directorio escogido [3]. Esta función convierte la interfaz en una herramienta efectiva de creación de datos.

Si el usuario solicita N placas, el sistema construye un conjunto

$$\mathcal{D}_N = \{(x_i, y_i)\}_{i=1}^N,$$

donde cada x_i es una imagen generada y y_i puede deducirse de la placa sintética de origen. A diferencia de una colección artesanal de ejemplos, aquí el proceso es escalable y reproducible. Esto tiene consecuencias directas en investigación: permite repetir experimentos, construir particiones controladas y analizar cómo cambia el desempeño al crecer el conjunto de entrenamiento.

También tiene consecuencias en docencia. Un profesor puede generar en minutos decenas o cientos de ejemplos para una práctica de clasificación, segmentación o lectura automática de caracteres. Incluso puede pedir a distintos grupos que entrenen modelos con datasets generados bajo condiciones visuales diferentes y luego comparar los resultados. En ese sentido, el sistema no solo fabrica imágenes; también fabrica escenarios pedagógicos.

6 Generación sintética y realista de matrículas vehiculares para entrenamiento de algoritmos de reconocimiento de patrones

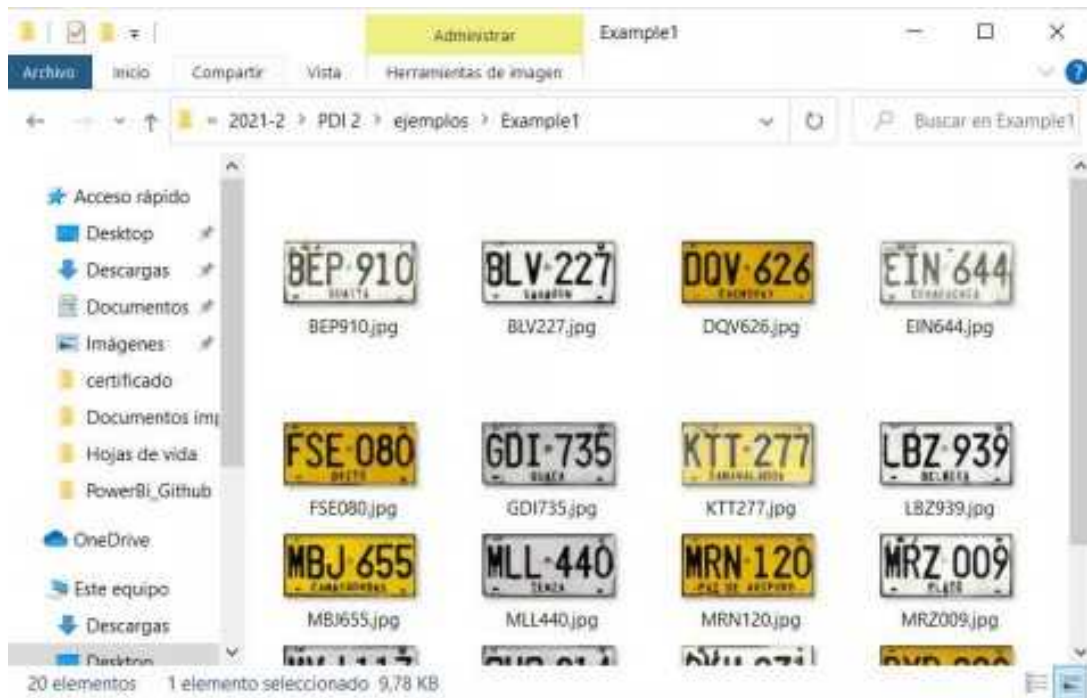


Figura 6.4: Proceso de generación masiva y carpeta de salida con las imágenes producidas por el sistema.

6.9. Arquitectura de software

El manual técnico indica que el sistema fue implementado con Python 3.9.7, TensorFlow 2.7.0, OpenCV, PySimpleGUI y Pillow, entre otras dependencias [4]. Además, documenta funciones y archivos como `gen_utils`, `pix2pixLPGe`, el método `unet_generator()`, el procedimiento `load_weights()`, la función `generate_images()` y la rutina principal de interfaz `main()` [4, 6].

Una representación compacta de la arquitectura puede darse mediante la tupla

$$\mathcal{A} = (\mathcal{I}, \mathcal{S}, \mathcal{G}, \mathcal{M}, \mathcal{E}),$$

donde:

- \mathcal{I} = interfaz gráfica de usuario,
- \mathcal{S} = generador sintético de placas,
- \mathcal{G} = modelo generativo entrenado,
- \mathcal{M} = módulo de modificadores visuales,
- \mathcal{E} = rutinas de exportación y generación masiva.

Esta formulación ayuda a leer el sistema como una cadena organizada y no como una aplicación monolítica. La interfaz recoge parámetros; el generador sintético fija el contenido; el modelo generativo aproxima el realismo; los modificadores expanden la variación; y el módulo de exportación produce el dataset utilizable. El manual técnico propone incluso un procedimiento para añadir nuevos métodos, ilustrado con la incorporación de una opción de rotación [4]. Esto confirma que el desarrollo fue pensado con modularidad.

6.10. Interfaz, extensibilidad y valor para enseñanza

La interfaz documentada en los manuales organiza el sistema de forma clara: en la parte superior izquierda se ingresan el texto de la placa y la ciudad; en la parte superior derecha se configura la generación aleatoria; al centro se observan la placa sintética y la placa alterada; y en la parte inferior se ubican los modificadores visuales [3]. Esta organización no solo es amigable, sino también pedagógica. Permite ver con claridad qué parte del proceso controla el usuario y cuál produce el modelo.

Esa claridad visual es importante para la docencia. Un estudiante puede comprender que existe una diferencia entre construir la placa base, pasarla por una red generativa y luego alterarla con parámetros adicionales. Esa intuición facilita discusiones sobre datasets, dominio sintético, dominio real, transferencia de estilo y robustez del modelo de reconocimiento.

El manual técnico refuerza esta idea al mostrar cómo añadir un nuevo método de rotación al sistema, desde la interfaz hasta la función de transformación correspondiente [4]. Ese detalle convierte al software en un objeto útil también para la enseñanza de ingeniería de software: no solo sirve para usar datasets, sino para mostrar cómo se extiende una herramienta modular con nuevos filtros y transformaciones.



Figura 6.5: Extensibilidad del sistema mediante la incorporación de nuevos filtros y transformaciones sobre las imágenes generadas.

6.11. Discusión

El desarrollo tiene un valor particular dentro del conjunto de trabajos del seminario porque actúa como infraestructura para otros proyectos. Mientras algunos software resuelven un problema final de medición o clasificación, este genera los datos con los que otros modelos pueden aprender. En ese sentido, funciona como una herramienta meta-metodológica: no

reconoce placas, pero contribuye a construir mejores reconocedores.

Su aporte en clases también merece destacarse. En formación en visión artificial y aprendizaje automático, no siempre es fácil conseguir datasets propios, ni mucho menos conjuntos suficientemente controlados para mostrar efectos específicos. El sistema llena ese vacío. Permite preparar ejemplos donde solo cambia un atributo visual, montar prácticas de entrenamiento con distintos niveles de dificultad y discutir con evidencia el impacto de la calidad del dato en el desempeño del modelo.

La limitación natural del sistema es que, por muy realistas que sean las imágenes generadas, siguen perteneciendo a un dominio inducido por el generador y por el modelo entrenado. Eso significa que no reemplazan por completo la necesidad de validación con datos reales. Precisamente ahí reside también una de sus mayores fortalezas didácticas: mostrar que la generación sintética puede ser muy potente, aunque siempre deba contrastarse con el dominio real.

6.12. Conclusiones

Este capítulo presentó un desarrollo orientado a la generación sintética y realista de matrículas vehiculares para la construcción de datasets útiles en reconocimiento automático. El sistema combina una placa sintética controlada, una transformación basada en CycleGAN y modificadores visuales que expanden la variabilidad del resultado.

Se mostró que el software no solo produce imágenes plausibles, sino que también permite generación masiva y controlada, lo cual lo convierte en una herramienta útil para el entrenamiento de CNNs, modelos clásicos de *machine learning* y algoritmos de procesamiento de imágenes aplicados a reconocimiento de placas.

Desde la perspectiva de la docencia, el desarrollo hace visible la relación entre dataset y modelo. Permite enseñar con ejemplos propios, comparar escenarios de entrenamiento y mostrar que el diseño de los datos es parte central del aprendizaje automático. En el marco del Seminario GeoLimna–GEPAR, este capítulo ejemplifica cómo un problema aparentemente auxiliar puede convertirse en una contribución estratégica para investigación, formación y prototipado tecnológico.

Agradecimientos

Se agradece a la Universidad de Antioquia, al grupo de investigación GEPAR y al laboratorio de Telemática de la Facultad de Ingeniería por el entorno académico y técnico que hizo posible este desarrollo. Se reconoce además la contribución de los autores consignados en el registro DNDA y el uso institucional certificado del software como herramienta para formación, investigación y experimentación aplicada [2, 7].

Referencias

- [1] D. S. Fernández Mc Cann y F. d. J. Vélez Macías, «Solicitud para asignación de puntos por desarrollo de software: LPGE, Licence Plate Generator,» Universidad de Antioquia, Medellín, Colombia, Documento institucional, 3 de abr. de 2023.
- [2] G. A. Marín Lopera, D. S. Fernández Mc Cann y C. A. Arbeláez Acevedo, *LPGE: License Plate Generator*, Registro ante la Dirección Nacional de Derecho de Autor, Libro 13, Tomo 91, Partida 345, fecha de registro: 26-ago-2022, 2022.
- [3] Universidad de Antioquia, *LPGe: License Plate Generator. User's Guide*, Universidad de Antioquia, Medellín, Colombia, 2022.
- [4] Universidad de Antioquia, *LPGe: License Plate Generator. Service Manual*, Universidad de Antioquia, Medellín, Colombia, 2022.
- [5] J.-Y. Zhu, T. Park, P. Isola y A. A. Efros, «Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks,» en *Proceedings of the IEEE International Conference on Computer Vision*, 2017.
- [6] P. Isola, J.-Y. Zhu, T. Zhou y A. A. Efros, «Image-to-Image Translation with Conditional Adversarial Networks,» en *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.
- [7] G. A. Marín Lopera y D. S. Fernández Mc Cann, «Certificación de uso del software LPGE,» Universidad de Antioquia, Medellín, Colombia, Certificación institucional, 4 de abr. de 2023.

Capítulo 7

Gestión y despliegue parametrizable de información de grupos de investigación mediante una plataforma web responsiva

Autores

Juan Alejandro Ríos Pérez¹; Gustavo Andrés Marín Lopera²; David Stephen Fernández Mc Cann³.

Resumen

La producción de un grupo de investigación no se agota en los artículos, prototipos o proyectos que realiza; también depende de su capacidad para mostrar, organizar y mantener visible esa producción ante estudiantes, pares, aliados y público general. En este capítulo se presenta un desarrollo web orientado a resolver ese problema: una plataforma de gestión y despliegue de contenido que permite publicar proyectos e integrantes de grupos de investigación a través de interfaces gráficas responsivas, personalizables y administrables sin necesidad de intervenir directamente el código fuente del sitio. El sistema integra un frontend en React, un backend tipo API REST construido con Node.js y Express, un me-

¹Estudiante de Ingeniería de Sistemas.

²Profesor y coordinador del laboratorio de Telemática.

³Profesor, Departamento de Ingeniería Electrónica y Telecomunicaciones.

^{01,2,3}Universidad de Antioquia. Grupo de investigación GEPAR.

canismo de caché para optimizar tiempos de respuesta y una estrategia de parametrización basada en hojas de cálculo de Google que facilita la actualización de contenido e identidad visual. Esta combinación convierte al desarrollo en una infraestructura ligera de presencia digital para grupos académicos. El capítulo reconstruye el problema institucional que dio origen al sistema, formaliza su arquitectura funcional, explica la lógica de parametrización y discute su valor como herramienta reusable para investigación, visibilidad científica y administración distribuida de contenido.

Palabras clave: sistema de gestión de contenidos; grupos de investigación; interfaz web; React; Node.js; API REST; parametrización; hojas de cálculo de Google.

7.1. Introducción

La vida de un grupo de investigación no depende únicamente de los resultados que produce, sino también de la forma en que esos resultados pueden organizarse, narrarse y ponerse en circulación. Este capítulo trata sobre un desarrollo de ese tipo. No mide una variable ambiental, no controla un robot y no clasifica imágenes microscópicas; su función es convertir la información dispersa de un grupo de investigación en una presencia digital estructurada, navegable y actualizable.

La documentación del software describe GROCMAN-WEB como un sistema de gestión de contenidos que permite mostrar información de proyectos y participantes de grupos de investigación a través de diferentes interfaces gráficas, con acceso a detalles como datos de contacto, biografías, descripciones de proyectos, videos y manuales [1, 2, 3, 4]. El sistema resuelve así el paso entre almacenar información y hacerla pública de forma útil.

Ese paso no siempre es trivial. En muchos grupos, la información existe, pero permanece fragmentada en carpetas, repositorios, presentaciones, hojas de cálculo o archivos internos que no ofrecen una presentación adecuada al público general. El manual técnico lo señala con claridad al indicar que GEPAR contaba con información en Google Drive, pero no con una página desde la cual cualquier persona pudiera conocer los proyectos, integrantes e información general del grupo [4]. El problema no era la ausencia de contenido, sino la ausencia de una forma operativa de exposición.

La originalidad del desarrollo reside en la manera de resolver ese problema. En lugar de obligar al usuario a editar plantillas web, tocar el backend o aprender desarrollo frontend, el sistema traslada la gestión de contenido y de parametrización a hojas de cálculo de Google, permitiendo una administración intuitiva y distribuida [1, 3, 4]. Esa decisión desacopla despliegue e ingreso de datos, facilita la sostenibilidad del sitio y evita que la edición del contenido quede restringida al equipo desarrollador.

7.2. Problema institucional y formulación general

El problema que da origen al sistema puede formularse así: dado un conjunto de proyectos, participantes y recursos asociados a un grupo de investigación, se requiere una plataforma web que permita organizar, presentar y actualizar esa información de manera pública, responsiva y fácilmente administrable por usuarios no expertos en programación.

Si se define el conjunto de proyectos como

$$\mathcal{P} = \{p_1, p_2, \dots, p_M\},$$

y el conjunto de integrantes como

$$\mathcal{U} = \{u_1, u_2, \dots, u_N\},$$

entonces el sistema debe construir una representación navegable donde cada proyecto p_i pueda asociarse a una descripción, un conjunto de participantes, un video y un manual, mientras cada integrante u_j pueda relacionarse con sus datos de contacto, su biografía y los proyectos en los que participa.

Desde esta perspectiva, el sistema no opera solamente sobre listas. Opera sobre relaciones. Una forma simple de expresarlo es mediante una aplicación

$$R : \mathcal{U} \times \mathcal{P} \rightarrow \{0, 1\},$$

donde $R(u_j, p_i) = 1$ si el integrante u_j participa en el proyecto p_i . Esta relación permite construir las páginas cruzadas del sitio: desde un autor hacia sus proyectos y desde un proyecto hacia sus integrantes.

La necesidad institucional de resolver este problema está bien documentada. El manual técnico y la OTRI señalan que el sistema fue pensado para ofrecer una plataforma pública desde la cual cualquier persona pudiera conocer información del grupo, sus proyectos y sus integrantes, y que además pudiera adaptarse a otros grupos de investigación de diferentes instituciones [4, 5]. Este último punto es importante porque el desarrollo no se limita a una página para GEPAR, sino que nace con vocación de generalización.

7.3. Más que una página web: una infraestructura ligera de visibilidad científica

Una página web estática organiza contenido. Un sistema como este organiza además el flujo de actualización, la presentación diferenciada por tipo de entidad y la personalización institucional. Esta diferencia es la que vuelve relevante el desarrollo dentro del seminario.

El manual de usuario explica que la página de inicio muestra barra de navegación, logos institucionales, una sección “acerca del grupo”, un listado de integrantes clasificados por tipo y una lista paginada de proyectos recientes. Además, el usuario puede hacer clic sobre un integrante o un proyecto para acceder a sus respectivas páginas de detalle [3]. Esta estructura responde a una arquitectura de representación bien definida.

Si se denota la interfaz principal por

$$\mathcal{I}_{home} = (\mathcal{N}, \mathcal{A}, \mathcal{U}', \mathcal{P}', \mathcal{F}),$$

entonces \mathcal{N} representa la barra de navegación, \mathcal{A} el bloque “acerca de”, \mathcal{U}' el subconjunto de integrantes visibles, \mathcal{P}' el subconjunto paginado de proyectos y \mathcal{F} el footer con redes sociales e identidad institucional. Esta formulación hace explícito que la página de inicio es una composición modular de componentes con función específica.

La misma lógica aparece en las vistas de detalle. La página de un integrante articula información personal, biografía y proyectos asociados. La página de proyecto articula un video, una descripción, los participantes y un enlace al manual de usuario [3]. El sistema transforma así entidades académicas dispersas en objetos digitales navegables.

7.4. Arquitectura de alto nivel

El manual técnico indica que el software fue creado de manera modular y que consiste en dos componentes de alto nivel: un frontend y un backend tipo API REST. El frontend fue desarrollado en React y el backend usando Express sobre Node.js [4]. Esta decisión tecnológica facilita separación de responsabilidades, reutilización de componentes y escalabilidad del sistema.

Una representación compacta de esa arquitectura puede darse mediante la tupla

$$\mathcal{A} = (\mathcal{F}, \mathcal{B}, \mathcal{C}, \mathcal{D}, \mathcal{K}),$$

donde:

\mathcal{F} = frontend responsivo en React,

\mathcal{B} = backend API REST en Node.js/Express,

\mathcal{C} = módulos de caché,

\mathcal{D} = fuentes de datos externas basadas en spreadsheets,

\mathcal{K} = capa de parametrización del sitio.

Esta arquitectura organiza el sistema como un flujo de procesamiento y publicación. El backend consulta y transforma datos; el frontend los presenta como interfaz navegable; la caché reduce latencia; y la parametrización desacopla identidad visual y contenido estructural del código mismo. El resultado es una plataforma ligera, pero conceptualmente bien resuelta.

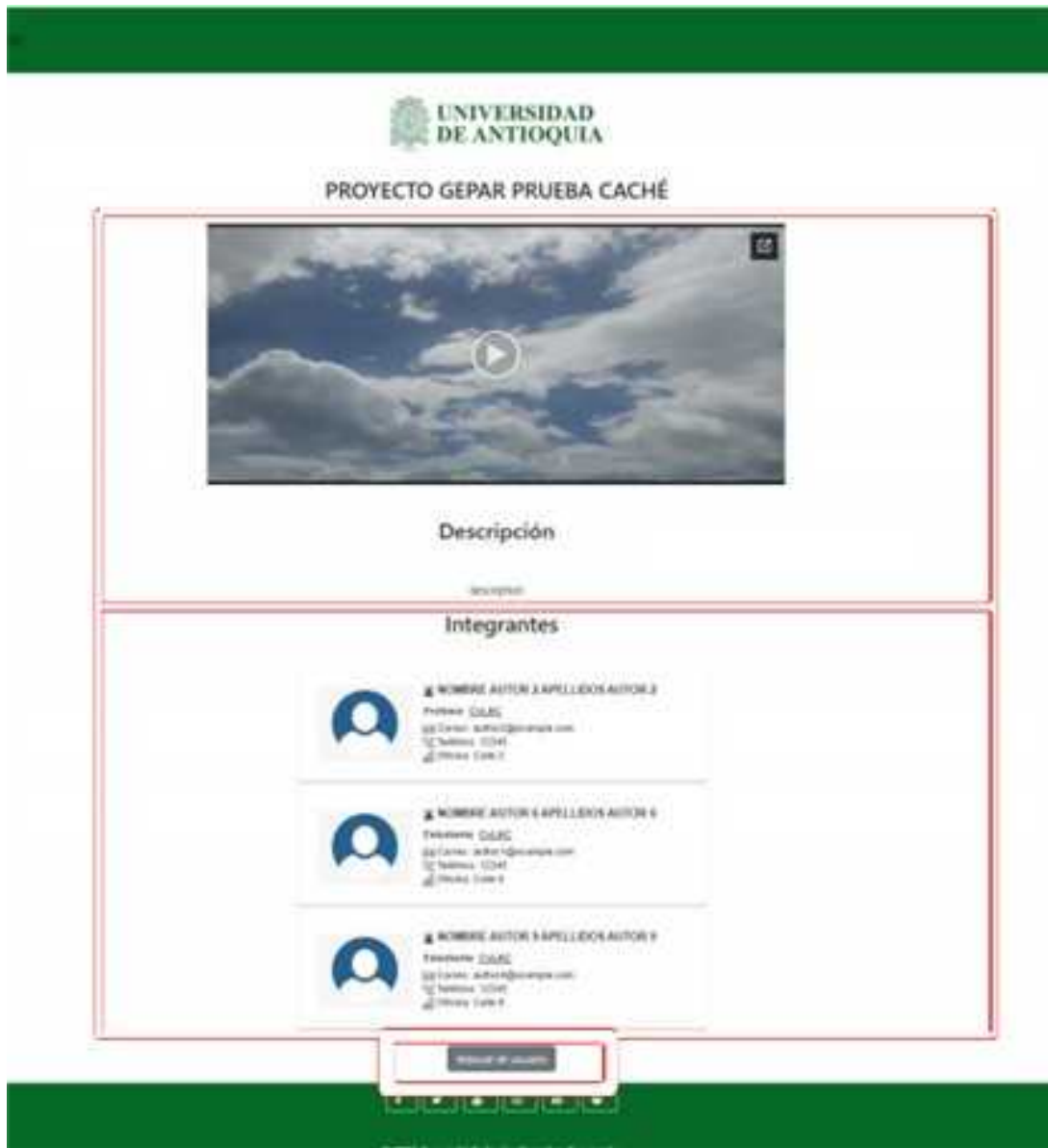


Figura 7.1: Arquitectura lógica y flujo de actualización del sistema, mostrando la relación entre hojas de cálculo, backend, control de caché, API REST, frontend y sitio desplegado.

7.5. Frontend responsivo y experiencia de navegación

La interfaz de GROCMAN-WEB fue desarrollada en React, una biblioteca de JavaScript orientada a la construcción de componentes reutilizables [4]. Esta decisión se refleja en la estructura del proyecto, donde el manual técnico muestra componentes como `Footer.js`, `NavBar.js`, `About.js`, `Author.js`, `AuthorCell.js`, `HomePage.js`,

`Project.js`, `ProjectCell.js` y `ProjectContainer.js` [4].

Desde el punto de vista de diseño, esto significa que la interfaz no fue pensada como una página rígida, sino como un conjunto de piezas reensamblables. Esta modularidad resulta especialmente importante para un sistema que quiere ser parametrizable y reusable. Si una institución desea adaptar el sitio a otro grupo de investigación, no necesita reconstruir toda la lógica visual; puede intervenir configuraciones y reutilizar la base estructural.

El manual de usuario describe tres vistas principales: la página de inicio, la página de integrante y la página de proyecto [3]. La primera organiza el acceso general al grupo; la segunda individualiza a cada participante; la tercera convierte cada proyecto en una entidad con video, descripción, equipo y documentación. Esta triplete de vistas resume la intención del sistema: pasar de un inventario plano a una narrativa navegable del grupo de investigación.



Figura 7.2: Página de inicio del sistema, con barra de navegación, sección “acerca de”, listado de integrantes, lista paginada de proyectos y footer con redes sociales.

7.6. Backend, API REST y acceso estructurado a la información

Mientras el frontend organiza la experiencia visual, el backend resuelve la obtención y estructuración de los datos. El manual técnico describe módulos como `cacheInit`, `cacheManagement`, `projectsGeneralController`, `Uncache` y `spreadsheetId`, además de las rutas HTTP desde las cuales el frontend consume información [4].

Esto puede representarse mediante una función general de servicio

$$S : \mathcal{Q} \rightarrow \mathcal{J},$$

donde \mathcal{Q} es el conjunto de consultas que llegan al backend y \mathcal{J} el conjunto de respuestas serializadas en JSON. En términos prácticos, el backend actúa como traductor entre la estructura de almacenamiento de los datos y las necesidades del frontend.

Las rutas documentadas incluyen accesos como

$$\left\{ \begin{array}{l} /projects, \\ /project/:id, \\ /projects/:offset/:limit, \\ /authordetails/:id, \\ /authors, \\ /frontconfiguration \end{array} \right\},$$

lo que muestra que la API no entrega simplemente archivos completos, sino recursos semánticamente diferenciados [4]. Esto tiene dos implicaciones. Primero, mejora la claridad del sistema. Segundo, facilita que otras interfaces o extensiones futuras puedan reutilizar el backend como fuente estructurada de información.

7.7. Parametrización como principio de diseño

Uno de los rasgos más distintivos del sistema es que la personalización del sitio se desplaza hacia hojas de cálculo de Google. El manual de usuario y el manual técnico explican que la aplicación usa dos spreadsheets clave: una para el manejo de caché y otra para la configuración del frontend [3, 4]. A partir de ellas se controlan títulos, colores, logos, redes sociales, textos de secciones y otros elementos de la interfaz.

7 Gestión y despliegue parametrizable de información de grupos de investigación mediante una plataforma web responsiva

Si se denota por

$$\Theta = \{\theta_1, \theta_2, \dots, \theta_L\}$$

el conjunto de parámetros de personalización, entonces el sitio desplegado puede verse como una función

$$W = \Psi(\mathcal{D}, \Theta),$$

donde \mathcal{D} representa los datos del grupo y Θ la configuración visual y semántica del frontend. Esta expresión ayuda a entender por qué el sistema puede adaptarse a distintos grupos sin cambiar su arquitectura base: basta con modificar datos y configuración.

Las tablas del manual de usuario muestran esta filosofía de manera explícita. Allí aparecen parámetros como `name`, `iconUrl`, `backgroundColor`, `navbarColor`, `institutionLogo`, `groupLogo`, `footerColor`, títulos de secciones, atributos de usuarios, textos de biografía, número de proyectos por página y textos de las vistas de proyecto [3]. Este nivel de parametrización convierte al sistema en una plantilla viva de presencia institucional, no en un sitio fijo.

<code>banner</code>	Logo de la aplicación que se ve en el inicio		<code>paginaDeInicio</code>	Cantidad de ítems de recursos, empezando en 2	2	<code>userType1InstitutionName</code>	Nombre del artículo 1 de los integrantes	<code>logro1</code>	Proyecto Geopar último 5 años
<code>backgroundColor</code>	Código de fondo de la aplicación, en el formato color hexadecimales		<code>paginaType1</code>	El título que llevará la sección de los primeros integrantes que se muestran, en este ejemplo profesores, en plural	Profesores	<code>proposeProjectPage</code>	Contenido de propuestas que se muestran en una ventana, en el formato 1. Con los botones que se muestran en el ejemplo se pueden ver los demás proyectos.		Proyecto Geopar último 5 años
<code>bannerColor</code>	Código de la barra de navegación		<code>paginaType2</code>	El título que llevará la sección de los siguientes integrantes que se muestran, en este ejemplo estudiantes, en plural	Estudiantes	<code>projectArticlesCount</code>	Cantidad de artículos de un proyecto, empezando en 3		Participantes ▲ Contiene autor 2 especifica autor 3 ▲ Contiene autor 4 especifica autor 5 ▲ Contiene autor 6 especifica autor 5 7/11 COMPLETOS
<code>bannerTitle</code>	Título de la barra de navegación		<code>paginaType3</code>	El título que llevará la sección de los primeros integrantes que se muestran, en este ejemplo profesor, en singular	Profesor	<code>projectArticleImage1</code>	Logo del primer artículo, en este caso de Geopar en blanco		Participantes
<code>institutionLogo</code>	Logo de la institución a la que pertenece el grupo de investigación		<code>paginaType4</code>	El título que llevará la sección de los siguientes integrantes que se muestran, en este ejemplo estudiantes, en singular	Estudiante	<code>projectArticleImage2</code>	Título del segundo artículo		Participantes
<code>groupLogo</code>	Logo del grupo de investigación		<code>headerLogo</code>	Logo que aparece al lado del nombre del usuario		<code>projectArticleImage3</code>	Logo del segundo artículo, en este caso de Geopar en blanco		Participantes
<code>headerColor</code>	Código de fondo del header		<code>userAvatarName</code>	Texto en qué se ve el enlace de perfil de usuario en este ejemplo todos tienen nombre COLAC	COLAC	<code>projectArticleImage4</code>	Logo del tercer artículo		Participantes
<code>background</code>	Código de fondo del grupo de investigación		<code>userTypeArticlesCount</code>	Cantidad de artículos de los integrantes, empezando en 4	4	<code>projectArticleImage5</code>	Logo del primer artículo		Participantes
<code>header</code>	Código de header del grupo de investigación		<code>userType1ArticleImage1</code>	Logo del artículo 1 de los integrantes		<code>projectArticleImage6</code>	Título del primer artículo		Participantes
<code>sidebar</code>	Código de sidebar del grupo de investigación		<code>userType1ArticleImage2</code>	Logo del artículo 2 de los integrantes		<code>projectArticleImage7</code>	Logo del segundo artículo		Participantes
<code>instagram</code>	Código de instagram del grupo de investigación		<code>userType1ArticleImage3</code>	Logo del artículo 3 de los integrantes		<code>projectArticleImage8</code>	Logo del tercer artículo		Participantes
<code>twitter</code>	Código de twitter del grupo de investigación		<code>userType1ArticleImage4</code>	Nombre del artículo 1 de los integrantes	Nombre del artículo 1 de los integrantes	<code>projectArticleImage9</code>	Título del segundo artículo, en este caso de Geopar en blanco		Participantes
<code>spotify</code>	Código de spotify del grupo de investigación		<code>userType1ArticleImage5</code>	Nombre del artículo 2 de los integrantes	Nombre del artículo 2 de los integrantes	<code>projectArticleImage10</code>	Logo del tercer artículo		Participantes
<code>copyright</code>	Texto de copyright del footer		<code>userType1ArticleImage6</code>	Nombre del artículo 3 de los integrantes	Nombre del artículo 3 de los integrantes	<code>projectArticleImage11</code>	Título del tercer artículo, en este caso de Geopar en blanco		Participantes
<code>aboutTitle</code>	Título de la sección ACERCA de la página de inicio	ACERCA DE GEPAR	<code>userType1ArticleImage7</code>	Nombre del artículo 4 de los integrantes, para este ejemplo se dejó en blanco	Nombre del artículo 4 de los integrantes, para este ejemplo se dejó en blanco	<code>authorBiographyText</code>	Texto de la página de autor en la parte de biografía		Biografía
<code>aboutDescription</code>	Título de la sección descripción de la	Geopar descripción							

Figura 7.3: Conjunto de componentes personalizables del frontend, gestionados desde una hoja de cálculo sin intervención directa sobre el código.

7.8. Caché como mecanismo de estabilidad y actualización controlada

La existencia de una capa de caché resuelve un problema importante: cómo permitir que los cambios hechos por administradores en hojas de cálculo se reflejen en el sitio sin

sobrecargar continuamente el servidor con consultas externas.

El manual técnico documenta métodos de inicialización, reinicio y comprobación de cachés, así como una lógica basada en claves de versión almacenadas en spreadsheets [4]. El manual de usuario explica, además, que para reflejar cambios en la configuración del frontend es necesario modificar el valor de `keyConfig`, y que el cambio se ve reflejado como máximo a los dos minutos [3].

Esto puede formalizarse de manera simple. Sea $k_c(t)$ el valor de la clave de configuración observada por el servidor en el tiempo t , y sea \hat{k}_c la versión local cacheada. Entonces el sistema activa una reinicialización cuando

$$k_c(t) \neq \hat{k}_c.$$

En términos operativos, esto significa que la actualización no depende de reiniciar manualmente la aplicación ni de editar código, sino de alterar una clave de control. El efecto institucional es relevante: usuarios administradores pueden actualizar contenido e identidad visual con un mecanismo comprensible y sin fricción técnica innecesaria.

7.9. Acceso a datos mediante hojas de cálculo

La decisión de usar hojas de cálculo de Google como backend liviano es uno de los rasgos más distintivos del sistema. Tanto el manual de usuario como el técnico explican que el sistema obtiene datos desde spreadsheets asociadas a proyectos, autores, detalles de autores, configuración del front y claves de caché [3, 4]. La OTRI resalta esta misma idea al señalar que la administración de información puede hacerse mediante herramientas simples como hojas de cálculo, sin requerir conocimientos avanzados de desarrollo web [5].

Si cada spreadsheet se denota por S_i , el backend implementa una familia de funciones

$$\Gamma_i : S_i \rightarrow J_i,$$

donde J_i es una representación JSON utilizable por el sistema. Esta transformación desacopla el formato en el que un administrador humano edita los datos del formato en el que el frontend consume la información.

Desde el punto de vista de ingeniería, esta decisión no solo simplifica la administración. También mejora la transferibilidad. Otros grupos o instituciones pueden adoptar el sistema sin construir de inmediato un backend complejo de gestión de contenido. Basta con

respetar la estructura de las hojas base y desplegar la aplicación correctamente.

7.10. Madurez tecnológica, validación y potencial de expansión

La OTRI caracteriza el desarrollo como un resultado de I+D+i con TRL 6, es decir, validado en un entorno relevante [5]. En su análisis destaca tres aspectos: pruebas en entorno real para el grupo GEPAR, interfaz intuitiva y escalable, y alta capacidad de parametrización [5]. Estas observaciones permiten leer el sistema no solo como prototipo, sino como una solución ya funcional.

La misma ficha de la OTRI incluye recomendaciones de avance que también resultan importantes: validarlo en otros grupos de investigación, optimizar aún más la experiencia de usuario y fortalecer documentación y soporte técnico [5]. Esto sugiere que el software resolvió de manera convincente su problema base, pero todavía tiene espacio de crecimiento como plataforma transversal para instituciones académicas.

También conviene señalar que la OTRI describe el sistema como un activo de conocimiento asociado al proyecto de variables ambientales de la ciénaga de Ayapel [5]. Este vínculo muestra que incluso dentro de proyectos marcados por sensores, monitoreo e instrumentación, surgieron necesidades de infraestructura digital para organizar y presentar resultados, personas y desarrollos.

7.11. Discusión

El desarrollo resuelve un problema de infraestructura informacional que suele quedar relegado frente a algoritmos o dispositivos más visibles. Un grupo de investigación no solo produce conocimiento; también necesita administrarlo, narrarlo y ponerlo en circulación. Sin esa capa de organización pública, muchos resultados permanecen encapsulados.

El valor del sistema radica en haber resuelto ese problema con una combinación coherente de tecnologías contemporáneas y herramientas cotidianas. React y Node.js aportan una base robusta y mantenible. Las hojas de cálculo de Google aportan una capa de administración accesible. El sistema de caché aporta estabilidad y control. La parametrización aporta reusabilidad. En conjunto, el desarrollo muestra una forma madura de pensar software institucional.

También es un caso útil para docencia. Permite mostrar a los estudiantes que no toda

innovación digital depende de modelos complejos de IA. Una arquitectura limpia, modular, integradora y orientada al usuario puede resolver un problema real de alto impacto organizacional. En ese sentido, este capítulo amplía la idea de qué cuenta como innovación tecnológica en un grupo de investigación.

7.12. Conclusiones

Este capítulo presentó un desarrollo orientado a la gestión y despliegue parametrizable de información de grupos de investigación mediante una plataforma web responsiva. Se mostró que el sistema organiza proyectos, participantes, biografías, videos y manuales en una interfaz navegable, fácil de administrar y adaptable a diferentes contextos institucionales.

La arquitectura combina frontend en React, backend API REST en Node.js/Express, una capa de caché para controlar actualizaciones y una estrategia de parametrización basada en hojas de cálculo de Google. Esta combinación permite que la actualización del contenido y de la identidad visual no dependa de modificar directamente el código del sitio, lo que constituye uno de sus aportes principales.

Desde la perspectiva del Seminario GeoLimna–GEPAR, el capítulo muestra que la innovación tecnológica también puede consistir en construir infraestructuras ligeras de visibilidad científica. No todo avance consiste en medir, clasificar o automatizar un objeto físico. También puede consistir en crear la plataforma que permite a una comunidad académica mostrarse, organizarse y permanecer visible.

Agradecimientos

Se agradece a la Universidad de Antioquia, al grupo de investigación GEPAR y al laboratorio de Telemática de la Facultad de Ingeniería por el entorno académico y técnico que hizo posible el desarrollo de esta plataforma. Se reconoce además la contribución de los autores consignados en el registro DNDA, así como la validación institucional y el concepto de madurez tecnológica emitido por la OTRI [2, 5, 6].

Referencias

- [1] D. S. Fernández Mc Cann y F. d. J. Vélez Macías, «Solicitud para asignación de puntos por desarrollo de software: GROCMAN-WEB,» Universidad de Antioquia, Medellín, Colombia, Documento institucional, 4 de abr. de 2023.
- [2] G. A. Marín Lopera, D. S. Fernández Mc Cann y J. A. Ríos Pérez, *GROCMAN-WEB*, Registro ante la Dirección Nacional de Derecho de Autor, Libro 13, Tomo 93, Partida 147, fecha de registro: 06-ene-2023, 2022.
- [3] Universidad de Antioquia, *GROCMAN-WEB: Manual de usuario*, Universidad de Antioquia, Medellín, Colombia, 2023.
- [4] Universidad de Antioquia, *GROCMAN-WEB: Manual técnico*, Universidad de Antioquia, Medellín, Colombia, 2023.
- [5] Oficina de Transferencia de Resultados de Investigación, Universidad de Antioquia, «RI-0084-2024: Concepto sobre la existencia y funcionalidad del software GrocMAN-Web,» Universidad de Antioquia, Medellín, Colombia, Concepto técnico OTRI, 16 de sep. de 2024.
- [6] G. A. Marín Lopera y D. S. Fernández Mc Cann, «Certificación de uso del software GROC-MAN,» Universidad de Antioquia, Medellín, Colombia, Certificación institucional, 4 de abr. de 2023.

Capítulo 8

Clasificación asistida de corales en imágenes submarinas mediante visión artificial

Autores

Sebastián Guzmán Obando¹; Daniela Cuartas Marulanda²; David Stephen Fernández Mc Cann³.

Resumen

Los arrecifes de coral constituyen uno de los ecosistemas marinos más valiosos y, al mismo tiempo, más sensibles a las perturbaciones antrópicas y climáticas. Su seguimiento exige procesos de observación suficientemente finos para reconocer cambios en composición bentónica, salud ecológica y presencia relativa de distintas clases de coral, arena y algas. Sin embargo, una parte importante de ese trabajo sigue dependiendo del análisis manual de imágenes submarinas, una tarea lenta, fatigante y vulnerable a variabilidad entre observadores. En este contexto se desarrolla CORALID, un software orientado a asistir la identificación de corales a partir de imágenes, integrando una interfaz gráfica de escritorio, un flujo de selección de regiones de interés y un módulo de clasificación sustentado en una línea metodológica previa de visión artificial y aprendizaje profundo.

¹Estudiante de Maestría en Ingeniería de Telecomunicaciones.

²Estudiante de Ingeniería Electrónica.

³Profesor, Departamento de Ingeniería Electrónica y Telecomunicaciones.

^{01,2,3}Universidad de Antioquia. Grupos de investigación GeoLimna y GEPAR.

El sistema permite cargar una imagen, seleccionar un punto o región relevante, generar un recorte y obtener una estimación de clase expresada mediante un gráfico de barras. Este capítulo presenta el problema que dio origen al desarrollo, formaliza su arquitectura conceptual, describe la implementación del software y discute su valor como herramienta reusable para monitoreo, investigación y conservación marina. Asimismo, sitúa a CORALID dentro de la trayectoria metodológica más amplia que comparó descriptores clásicos y arquitecturas profundas sobre imágenes coralinas, mostrando que el desarrollo no es una interfaz aislada, sino la cristalización operativa de una línea de investigación con resultados experimentales consistentes.

Palabras clave: visión artificial; arrecifes coralinos; clasificación de imágenes; aprendizaje profundo; software científico; CORALID.

8.1. Introducción

El monitoreo de arrecifes coralinos depende cada vez más de bancos de imágenes submarinas. Capturar estas imágenes puede hacerse con relativa eficiencia mediante buzos, cámaras o vehículos subacuáticos; interpretarlas con consistencia, en cambio, exige tiempo experto, criterio ecológico y una atención visual difícil de sostener cuando el volumen de datos crece. La tesis asociada a este desarrollo señala precisamente que el análisis manual de dichas imágenes sigue siendo costoso en términos de tiempo y esfuerzo especializado [1].

CORALID surge en ese punto de tensión. No fue concebido como un clasificador abstracto desconectado del problema ecológico, sino como una herramienta de apoyo para facilitar la labor de identificación de corales en imágenes submarinas. La documentación institucional del software lo presenta como un sistema de visión artificial diseñado para identificación precisa y eficiente de corales mediante imágenes capturadas bajo el agua, con aplicación en estudios ecológicos, monitoreo y conservación [2]. Esta formulación delimita con claridad el alcance del desarrollo: no pretende reemplazar la interpretación biológica del experto, sino asistirla y hacerla más operativa.

El valor del software se entiende mejor cuando se lo sitúa dentro de una trayectoria metodológica mayor. El trabajo de maestría del que deriva no se limitó a programar una interfaz. Desarrolló una revisión del problema, estudió técnicas de análisis de texturas, evaluó clasificadores basados en máquinas de soporte vectorial y comparó arquitecturas profundas como SqueezeNet, ResNet152, DarkNet, DenseNet201 y VGG19 sobre imágenes de arrecifes coralinos [1, 3]. CORALID representa, en ese sentido, el paso que lleva una línea experimental hacia una herramienta usable. Esa transición es especialmente valiosa

en el marco del *Seminario GeoLimna-GEPAR*, donde los desarrollos no se valoran solo por su formulación, sino por su capacidad de convertirse en sistemas operativos, documentados y reutilizables.

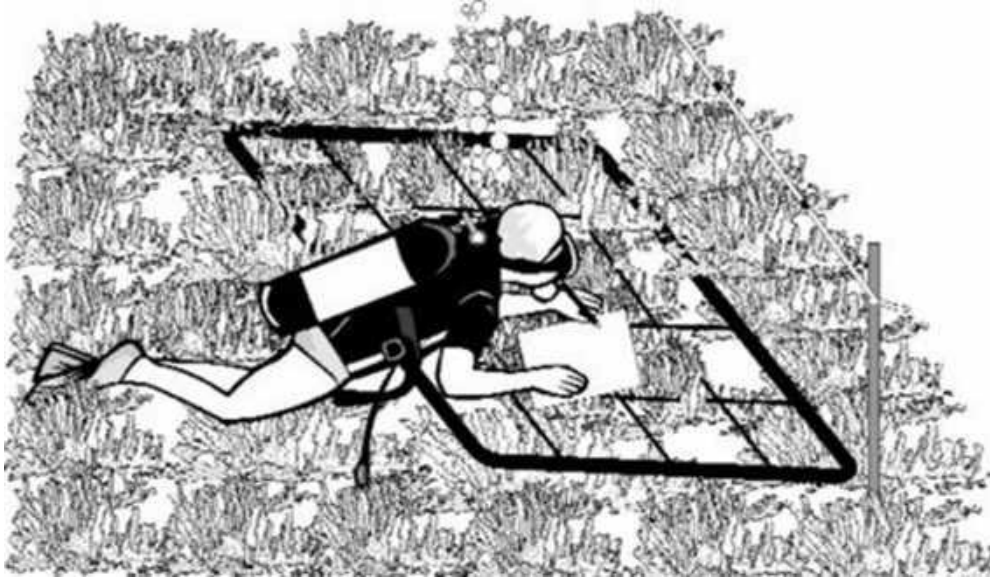


Figura 8.1: Escena submarina representativa del dominio ecológico sobre el cual opera el sistema de clasificación asistida.

8.2. Problema y justificación del desarrollo

El problema que da origen a CORALID puede formularse así: dado un conjunto de imágenes submarinas de arrecifes coralinos, se requiere un sistema que permita asistir la identificación de regiones de interés pertenecientes a diferentes clases bentónicas, reduciendo el esfuerzo del etiquetado manual y aumentando la trazabilidad del proceso de clasificación. En la práctica, esto significa convertir una escena visual compleja en una secuencia operativa simple para el usuario: cargar la imagen, ubicar el punto de interés, recortar la región relevante y obtener una estimación de clase con soporte visual.

Si se representa una imagen submarina por la función

$$I : \Omega \subset \mathbb{R}^2 \rightarrow \mathbb{R}^3,$$

entonces Ω representa el dominio espacial de la imagen y el codominio en \mathbb{R}^3 alude a su representación en color. Sobre esa imagen, el usuario selecciona una posición (x_0, y_0) a partir de la cual el sistema construye un recorte local. Si el recorte tiene dimensiones

$w \times h$, puede expresarse como

$$R_{(x_0, y_0)}(u, v) = I(x_0 + u, y_0 + v), \quad -\frac{w}{2} \leq u \leq \frac{w}{2}, \quad -\frac{h}{2} \leq v \leq \frac{h}{2}.$$

Esta expresión resume la lógica del sistema: no clasifica necesariamente la imagen completa de forma densa, sino una ventana de observación centrada en la región que el usuario considera relevante.

Esa decisión es coherente con la interfaz documentada. El manual de usuario muestra que el software organiza el proceso mediante tres acciones principales: *cargar imagen*, *seleccionar* y *clasificar*, apoyadas por opciones de visualización como *imagen original*, *zoom in* y *zoom out* [4]. La necesidad de este flujo es clara: las imágenes coralinas contienen heterogeneidad espacial, oclusiones, cambios de iluminación y regiones donde varias clases coexisten. Por ello, resulta razonable que la decisión inicial sobre dónde mirar siga dependiendo del usuario, mientras la decisión sobre a qué clase se parece el recorte sea asistida por el sistema.

La justificación del desarrollo aparece también con nitidez en la solicitud institucional de puntos por software, donde se afirma que CORALID busca automatizar parte del análisis tradicional de imágenes de coral, permitiendo incluso la intervención manual del usuario para ajustar y mejorar la predicción [2]. Esto revela una decisión de ingeniería madura: el sistema no fue pensado como una caja negra autosuficiente, sino como una herramienta híbrida donde el criterio humano y la inferencia computacional colaboran.

8.3. Fundamento metodológico: de la caracterización coralina al software

La utilidad de CORALID no descansa solo en la interfaz, sino en la base metodológica que le da sustento. Esa base fue desarrollada en el trabajo de maestría sobre caracterización de imágenes de arrecifes coralinos, donde se propusieron dos rutas de análisis. La primera consistió en trabajar con descriptores construidos a partir de filtros de Gabor y clasificadores SVM. La segunda, en entrenar diferentes topologías de redes neuronales convolucionales sobre clases coralinas y no coralinas usando la base de datos *Moorea Labeled Coral* [1, 5, 6].

Sea

$$\mathcal{D} = \{(X_i, y_i)\}_{i=1}^N$$

un conjunto de muestras de entrenamiento, donde X_i es un parche de imagen coralina e

y_i su clase asociada. El problema supervisado consiste en estimar una función

$$f_\theta : \mathcal{X} \rightarrow \mathcal{C},$$

tal que para una nueva muestra X se obtenga

$$\hat{y} = f_\theta(X),$$

donde θ representa el conjunto de parámetros aprendidos y \mathcal{C} es el conjunto de clases. En la línea experimental de CNN, las clases incluyeron *Acropora*, *Montipora*, *Pavona*, *Pocillopora*, *Porites*, algas coralinas incrustantes, *turf algae*, macroalgas y arena [1].

Como la base de datos presentaba desbalance entre clases, se introdujo una estrategia de aumento de datos. Si \mathcal{A}_j denota la transformación j aplicada a una muestra, el conjunto expandido puede escribirse como

$$\mathcal{D}^+ = \{(\mathcal{A}_j(X_i), y_i)\}_{i=1, \dots, N; j=1, \dots, m_i}.$$

Aquí m_i representa el número de transformaciones aplicadas a la muestra i . En términos prácticos, esta expansión se usó para fortalecer clases minoritarias y mejorar la generalización del entrenamiento. El sentido de esta formalización es importante para el capítulo: CORALID no nace de una ocurrencia de interfaz, sino de una ruta metodológica donde las clases, las transformaciones y la lógica de clasificación fueron definidas y evaluadas previamente.

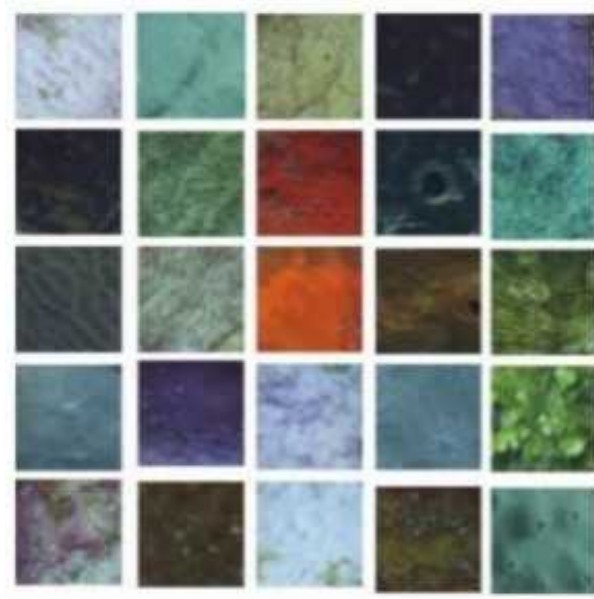


Figura 8.2: Ejemplos de recortes coralinos utilizados durante el entrenamiento y validación de la línea de clasificación subyacente al software.

8.4. Arquitectura conceptual de CORALID

Desde una perspectiva funcional, CORALID puede representarse como una cadena de cuatro etapas:

$$I \xrightarrow{\Phi_1} R \xrightarrow{\Phi_2} z \xrightarrow{\Phi_3} \hat{p} \xrightarrow{\Phi_4} \hat{y}.$$

La transformación Φ_1 corresponde al paso desde la imagen completa I hacia el recorte R definido por el usuario. La transformación Φ_2 corresponde al preprocesamiento necesario para que ese recorte tenga el formato de entrada que exige el clasificador. La transformación Φ_3 produce un vector de puntuaciones o probabilidades por clase. Finalmente, Φ_4 entrega la clase estimada.

Si el sistema devuelve una distribución sobre K clases, el vector de salida puede escribirse como

$$\hat{p} = [p_1, p_2, \dots, p_K], \quad \sum_{k=1}^K p_k = 1,$$

y la decisión final como

$$\hat{y} = \arg \max_{k \in \{1, \dots, K\}} p_k.$$

Esta ecuación explica por qué la salida del software aparece como un gráfico de barras. La interfaz no devuelve simplemente una etiqueta nominal; expone la distribución com-

parativa entre clases, permitiendo que el usuario vea si la decisión es muy marcada o si existen clases competidoras con puntajes cercanos.

La documentación de usuario confirma esta lógica: una vez generado el recorte, el botón de clasificación produce una visualización donde la clase con mayor porcentaje domina el gráfico [4]. Este detalle aporta trazabilidad, porque hace visible la estructura de la decisión computacional.

8.5. Interfaz de usuario y experiencia de operación

Un rasgo importante de CORALID es que la interfaz fue organizada siguiendo la propia secuencia cognitiva del usuario. En la ventana principal aparece primero un espacio de carga para la imagen completa; a su lado, un espacio para observar el recorte; debajo, un gráfico de clasificación; y alrededor, los controles necesarios para navegar por la imagen y ejecutar el flujo de trabajo. El manual de usuario documenta además que los botones de zoom deshabilitan temporalmente la selección de recuadro, forzando a que la clasificación se haga sobre la vista original cuando se requiera un recorte estable [4]. Esta restricción corresponde a una decisión de diseño para evitar inconsistencias geométricas en la selección.

Si se denota por s el estado de la interfaz, el comportamiento general puede describirse como una máquina de estados simple:

$$s_0 \xrightarrow{\text{chooseImage}} s_1 \xrightarrow{\text{onClick}} s_2 \xrightarrow{\text{cropImage}} s_3 \xrightarrow{\text{clasify}} s_4,$$

donde:

- s_0 : sin imagen cargada,
- s_1 : imagen cargada y visible,
- s_2 : región de interés señalada,
- s_3 : recorte generado,
- s_4 : clasificación desplegada.

Esta representación muestra que el software no se comporta como una ejecución monolítica, sino como una secuencia de estados progresivos. El usuario necesita saber siempre en qué punto del flujo se encuentra y qué acción habilita la siguiente.



Figura 8.3: Interfaz principal del sistema, con imagen cargada, recorte seleccionado, controles de navegación y gráfico de clasificación.

8.6. Arquitectura técnica del software

El manual técnico describe a CORALID como una aplicación modular compuesta por una capa visual y una capa de procesamiento. Entre las librerías empleadas aparecen `tkinter`, `PIL`, `cv2`, `keras`, `matplotlib` y `os`; entre las funciones principales se documentan `chooseImage()`, `onClick(event)`, `drawRectangle()`, `cropImage()`, `showCrop()` y `clasify()` [7]. El hecho de que estas funciones queden explicitadas en el manual sugiere que el software alcanzó un nivel de organización superior al del prototipo improvisado.

Una forma compacta de representar la arquitectura es mediante la tupla

$$\mathcal{A} = (\mathcal{U}, \mathcal{G}, \mathcal{C}, \mathcal{V}),$$

donde \mathcal{U} es el módulo de interacción con el usuario, \mathcal{G} el conjunto de elementos gráficos, \mathcal{C} el clasificador y \mathcal{V} el módulo de visualización de resultados.

A nivel operacional, el núcleo del programa puede entenderse como la composición

$$\text{clasify} \circ \text{cropImage} \circ \text{onClick} \circ \text{chooseImage}.$$

Esta expresión resume el orden lógico del sistema: elegir la imagen, detectar el punto de interés, generar el recorte y luego clasificar. Lo importante es que esta cadena coincide con la experiencia real del usuario. No hay disonancia entre arquitectura técnica e interacción.

El manual técnico explica además ejemplos para añadir botones, logos, canvas y nuevas funciones dentro de la interfaz [7]. Esto revela una cualidad valiosa en el contexto del seminario: el software no fue pensado como producto inmóvil, sino como una base extensible para nuevas funciones o variantes del sistema.

8.7. Requerimientos técnicos y condiciones de implementación

Conviene dejar consignadas las condiciones materiales en las que se desarrolló el software, porque ello permite valorar su grado de accesibilidad y reproducibilidad. El manual técnico reporta que el sistema fue construido sobre un equipo con procesador Intel Core i5-9300H, 8 GB de RAM, 480 GB de SSD, tarjeta gráfica NVIDIA GeForce GTX 1050 y Windows 10 de 64 bits. En cuanto al entorno de software, se utilizaron Python 3.9.12, Spyder dentro de Anaconda, TensorFlow 2.9.1, Pillow 9.0.1 y OpenCV 4.6.0 [7].

Esta información muestra que el sistema fue desarrollado bajo condiciones de cómputo razonables para un contexto universitario. Esto refuerza su valor como capacidad instalada: no se trata de un desarrollo que solo podría existir en infraestructuras excepcionales, sino de una herramienta realizable y mantenible en un entorno académico ordinario, aunque técnicamente serio.

8.8. Resultados experimentales que fundamentan la línea CORALID

La tesis de maestría asociada permite ver con claridad que la línea técnica subyacente al software alcanzó resultados importantes. Para el caso de redes neuronales convolucionales, se evaluaron cinco topologías principales. Si se define la exactitud sobre prueba como

$$\text{Acc}_{test} = \frac{N_{aciertos}}{N_{pruebas}},$$

entonces los resultados reportados fueron los siguientes:

Tabla 8.1: Desempeño de las arquitecturas CNN evaluadas en la línea experimental asociada a CORALID

Modelo	Accuracy Train (%)	Accuracy Test (%)
SqueezeNet	94.2	90.1
ResNet152	97.0	94.5
DarkNet	97.0	93.9
DenseNet201	86.2	82.9
VGG19	84.3	83.1

Estos resultados muestran una diferencia clara entre arquitecturas. ResNet152 obtuvo el mejor comportamiento en prueba con 94.5 %, seguida por DarkNet con 93.9 % y SqueezeNet con 90.1 % [1]. La lectura de estos resultados debe hacerse con cuidado. No significan que el software final implemente exactamente todas las variantes evaluadas, pero sí indican que la línea de clasificación coralina sobre la que se apoya CORALID alcanzó un nivel de madurez suficiente como para justificar su encapsulamiento en una herramienta operativa.

La tesis también destaca que SqueezeNet resulta atractiva en entornos con recursos más limitados, mientras que ResNet152 entrega la mejor exactitud cuando se dispone de mayor capacidad de cómputo [1]. Esta observación resulta útil para pensar la evolución futura del sistema: abre la posibilidad de versiones más livianas o más exigentes según el contexto de despliegue.

8.9. Protección intelectual, uso y valor como capacidad instalada

La madurez de CORALID no se expresa únicamente en su formulación técnica o en sus resultados experimentales. También aparece en su protección y uso institucional. El certificado de la DNDA registra *CORAL ID* como software original, con fecha 26 de julio de 2023, identificando a Sebastián Guzmán Obando, David Fernández Mc Cann y Daniela Cuartas Marulanda como autores, y a la Universidad de Antioquia como titular patrimonial y productor [8]. El documento describe la obra como un software de identificación de corales que recibe una imagen y da como salida la posible clase a la que puede pertenecer, orientado a facilitar la labor de etiquetado de los biólogos.

Por su parte, la certificación de uso emitida en junio de 2024 deja constancia de que el grupo GEPAR posee licencia de uso y emplea activamente el software en investigacio-

nes relacionadas con identificación y monitoreo de corales submarinos [9]. La solicitud institucional agrega además que el proyecto contribuyó a la formación de estudiantes y al fortalecimiento de capacidades del grupo en visión artificial aplicada a conservación marina [2].

En consecuencia, el desarrollo debe entenderse como algo más que una interfaz o un ejercicio de programación. Es una capacidad instalada que articula formación, desarrollo metodológico, protección intelectual y uso efectivo en investigación.

8.10. Discusión

CORALID representa con claridad una transición que en la universidad no siempre se logra con la misma consistencia: la transición desde una línea experimental hacia una herramienta usable. En ocasiones se produce una tesis valiosa, pero esa tesis no llega a software. En otras, se construye una interfaz sin un trasfondo metodológico suficientemente sólido. En este caso convergen ambas dimensiones. Existe una investigación de base, existen resultados cuantitativos, existen manuales técnico y de usuario, existe protección ante la DNDA y existe certificación de uso institucional. Esa convergencia es la que da al desarrollo su peso dentro del seminario.

También conviene subrayar que el sistema, tal como está documentado, no debe describirse como una plataforma de segmentación masiva completamente automática sobre escenas completas, sino como un sistema de clasificación asistida sobre recortes seleccionados por el usuario. Lejos de disminuir su valor, esta precisión lo vuelve más serio. Permite que el capítulo describa con fidelidad lo que el software efectivamente hace y cómo lo hace. Al mismo tiempo, deja abierta la posibilidad de una evolución futura hacia niveles mayores de automatización, aprovechando la base metodológica ya construida.

Por último, el carácter modular descrito en el manual técnico sugiere que el sistema puede seguir creciendo. Si ya organiza con claridad la carga de imagen, la selección, el recorte y la clasificación, entonces es razonable pensar en extensiones futuras como incorporación de nuevos modelos, nuevas clases, mejores mecanismos de navegación o integración con flujos más amplios de monitoreo. En ese sentido, el software no cierra una línea; la consolida.

8.11. Conclusiones

Este capítulo presentó a CORALID como un software de visión artificial orientado a asistir la identificación de corales a partir de imágenes submarinas mediante un flujo de selección

de regiones de interés y clasificación asistida. Se mostró que el desarrollo no se reduce a una interfaz gráfica, sino que se apoya en una trayectoria metodológica más amplia donde se evaluaron descriptores clásicos y arquitecturas profundas sobre imágenes coralinas.

La formulación matemática permitió dejar explícito que el sistema transforma una imagen completa en un recorte local y, a partir de él, en una distribución de probabilidades por clase. La descripción técnica mostró además que la implementación modular organiza con claridad interacción, navegación, recorte, clasificación y visualización. Los resultados experimentales reportados en la tesis, particularmente los de ResNet152 y SqueezeNet, aportan el sustento cuantitativo de la línea y justifican el paso desde el experimento hacia la herramienta operativa.

Desde la perspectiva del *Seminario GeoLimna–GEPAR*, CORALID ejemplifica de forma precisa lo que significa un desarrollo maduro: una pregunta ecológica real, una metodología formulada, resultados evaluados, una herramienta funcional, documentación técnica, uso institucional y posibilidad de continuidad. Esa combinación es la que convierte un trabajo universitario en memoria técnica perdurable.

Agradecimientos

Se agradece a la Universidad de Antioquia y a los grupos de investigación GeoLimna y GEPAR por el entorno académico, técnico e institucional que hizo posible el desarrollo de CORALID. Se reconoce de manera especial el trabajo de Sebastián Guzmán Obando y Daniela Cuartas Marulanda en la consolidación del software y su articulación con la línea de investigación en visión artificial aplicada a ecosistemas coralinos. Se agradece además la apropiación institucional del desarrollo y su uso en investigaciones orientadas a monitoreo y conservación marina [2, 9].

Referencias

- [1] S. Guzmán Obando, «Sistema para caracterización de imágenes de arrecifes coralinos mediante el desarrollo de un software de visión computacional,» Tesis de Maestría en Ingeniería de Telecomunicaciones, Universidad de Antioquia, Medellín, Colombia, 2024.
- [2] D. S. Fernández Mc Cann, *Solicitud para asignación de puntos por desarrollo de software: CORALID*, Documento institucional, Universidad de Antioquia, Medellín, Colombia, jun. de 2024.

- [3] S. Guzman Obando, D. Cuartas Marulanda y D. S. Fernández Mc Cann, «Image characterization of coral reefs using computer vision techniques and convolutional neural networks (CNNs),» en *Engineering for Transformation*, Editorial EIA, 2022, págs. 460-466.
- [4] *Coral ID: Manual de usuario*, Universidad de Antioquia, Medellín, Colombia, 2023.
- [5] O. Beijbom, P. J. Edmunds, D. I. Kline, B. G. Mitchell y D. Kriegman, «Automated Annotation of Coral Reef Survey Images,» en *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2012, págs. 1170-1177.
- [6] O. Beijbom et al., «Towards Automated Annotation of Benthic Survey Images: Variability of Human Experts and Operational Modes of Automation,» *PLOS ONE*, vol. 10, n.º 7, e0130312, 2015.
- [7] *Coral ID: Manual técnico*, Universidad de Antioquia, Medellín, Colombia, 2023.
- [8] Dirección Nacional de Derecho de Autor, *CORAL ID. Certificado de registro de soporte lógico–software*, Libro 13, Tomo 95, Partida 268. Fecha de registro: 26-jul-2023, 2023.
- [9] D. S. Fernández Mc Cann y J. P. Serna López, *Certificación de uso del software CORALID*, Certificación institucional, Universidad de Antioquia, Medellín, Colombia, jun. de 2024.
- [10] M. Modasshir, A. Q. Li e I. Rekleitis, «MDNet: Multi-Patch Dense Network for Coral Classification,» en *OCEANS 2018 MTS/IEEE Charleston*, 2018, págs. 1-6.
- [11] J. P. Pierce, Y. Rzhanov, K. Lowell y J. A. Dijkstra, «Reducing Annotation Times: Semantic Segmentation of Coral Reef Survey Images,» en *Global Oceans 2020: Singapore – U.S. Gulf Coast*, 2020, págs. 1-9.

Capítulo 9

Caracterización multitemporal de coberturas flotantes en embalses mediante procesamiento de imágenes satelitales multiespectrales

Autores

Ana María Marín Guerra¹; Juan Guillermo Quevedo²; Fabio de Jesús Vélez Macías³; David Stephen Fernández Mc Cann⁴.

Resumen

El seguimiento de coberturas flotantes en embalses exige metodologías capaces de integrar imágenes satelitales multiespectrales, criterios de clasificación espacialmente consistentes y salidas cuantitativas útiles para el monitoreo ambiental. En este capítulo se presenta un desarrollo metodológico y computacional orientado a la caracterización multitemporal de coberturas flotantes en el embalse del proyecto Hidroituango, a partir del procesamiento de imágenes satelitales PlanetScope de alta resolución espacial y espectral. La propuesta parte de imágenes ráster previamente corregidas y recortadas al área activa del espejo

¹Estudiante de Maestría en Ingeniería Ambiental.

²Estudiante de Ingeniería Electrónica.

³Profesor titular, Departamento de Ingeniería Ambiental y Sanitaria.

⁴Profesor titular, Departamento de Ingeniería Electrónica y Telecomunicaciones.

^{01,2,3,4}Universidad de Antioquia. Grupos de investigación GeoLimna y GEPAR.

de agua, y construye una cadena operativa que permite clasificar la superficie en cuatro categorías: agua, maderos flotantes, vegetación acuática y sedimentos. A partir de dicha clasificación se estima, para cada clase, el área ocupada en kilómetros cuadrados y su proporción relativa respecto del área evaluada. Más allá de su implementación computacional, el trabajo tiene valor metodológico porque traduce una necesidad real de gestión ambiental en un procedimiento reproducible, documentado y utilizable por investigadores y tomadores de decisión. El capítulo expone el problema que motivó el desarrollo, formaliza la arquitectura del proceso, describe la lógica de clasificación y discute su alcance dentro del monitoreo de materiales flotantes y del estudio espacio-temporal de su dinámica en cuerpos de agua regulados.

Palabras clave: teledetección; PlanetScope; embalses; coberturas flotantes; clasificación multiespectral; monitoreo ambiental; procesamiento digital de imágenes.

9.1. Introducción

En cuerpos de agua de gran extensión, la observación visual aislada no basta para seguir la dinámica de coberturas flotantes con continuidad y consistencia. La presencia de vegetación flotante, maderos arrastrados, sedimentos en superficie y otras formas de cobertura cambiante exige metodologías que permitan observar de manera sistemática, cuantificar con criterios homogéneos y comparar temporalmente lo observado. Ese es el problema al que responde el desarrollo documentado en este capítulo.

La documentación técnica del proyecto indica que el proceso fue concebido para el tratamiento de imágenes satelitales multiespectrales de la plataforma PlanetScope, provenientes de satélites Dove con resolución espacial de 3 m, resolución radiométrica de 16 bits, resolución temporal entre 1 y 3 días según condiciones atmosféricas y una resolución espectral de ocho bandas, desde *coastal blue* hasta *near-infrared* [1, 2]. Sobre esa base, el procedimiento busca caracterizar multitemporalmente coberturas flotantes presentes en el espejo de agua del embalse, clasificándolas en cuatro categorías operativamente relevantes: agua, maderos flotantes, vegetación acuática y sedimentos [1, 3, 4].

La relevancia del trabajo no radica únicamente en haber construido una aplicación funcional. Su valor principal está en el proceso que formaliza. En lugar de limitarse a mostrar una imagen clasificada, el desarrollo organiza una secuencia técnica que va desde la adquisición de datos multiespectrales hasta la obtención de métricas de ocupación espacial por clase. La certificación institucional de uso indica además que esta metodología está siendo empleada activamente en investigaciones relacionadas con la caracterización multitemporal de coberturas flotantes y que ha servido de apoyo al proyecto SIMEVAM y al

desarrollo de una maestría [5].

La propuesta de investigación asociada a la maestría de Ana María Marín Guerra ayuda a situar mejor el trasfondo científico del desarrollo. Allí se plantea que el embalse de Hidroituango presenta acumulación de materiales flotantes asociados al arrastre de residuos vegetales, proliferación de plantas acuáticas y, de manera ocasional, presencia de fitoplancton y cianobacterias. También se señala que, aunque las imágenes PlanetScope permiten calcular índices de vegetación, por sí solos estos no han sido suficientes para discriminar de manera robusta los distintos materiales flotantes presentes en el sistema [6]. En ese sentido, la metodología de clasificación aquí presentada debe entenderse como una respuesta aplicada a un problema ecológico y operativo concreto.

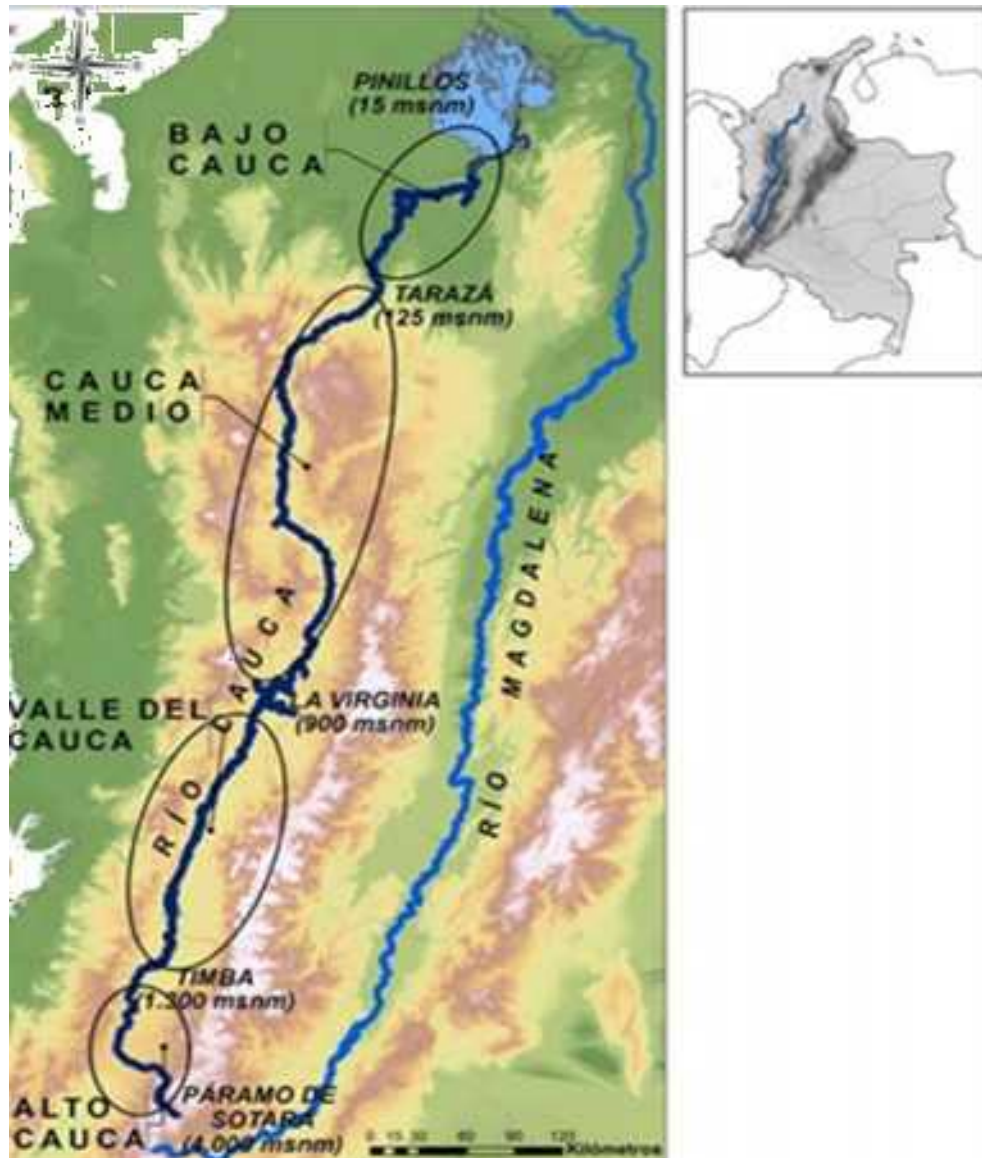


Figura 9.1: Vista representativa del embalse y del espejo de agua con presencia de materiales flotantes, correspondiente al dominio ambiental sobre el cual se aplica la metodología.

9.2. Problema y justificación del desarrollo

El problema puede formularse de la siguiente manera: dado un conjunto de imágenes satelitales multiespectrales de un embalse tomadas en diferentes fechas, se requiere una metodología que permita identificar, cuantificar y comparar temporalmente las coberturas flotantes presentes sobre el espejo de agua, de forma que la información obtenida sea útil para el monitoreo ambiental y la toma de decisiones.

Sea

$$\mathcal{I} = \{I_{t_1}, I_{t_2}, \dots, I_{t_T}\}$$

el conjunto de imágenes disponibles para un intervalo temporal dado, donde cada I_{t_k} corresponde a una escena adquirida en el instante t_k . Cada imagen está compuesta por ocho bandas espectrales:

$$I_{t_k}(x, y) = [b_1(x, y), b_2(x, y), \dots, b_8(x, y)],$$

donde $b_j(x, y)$ representa la reflectancia o intensidad registrada en la banda j para el píxel ubicado en (x, y) .

La metodología no opera sobre toda la escena satelital indiscriminadamente. Según el manual de usuario, el área activa de procesamiento corresponde únicamente a la porción previamente recortada al espejo de agua; el resto del rectángulo visualizado, mostrado en negro, no participa en el cálculo de áreas ni porcentajes [1]. Por ello, si se define $\Omega_w \subset \Omega$ como el dominio espacial del espejo de agua, entonces la imagen efectiva de trabajo se restringe a

$$I_{t_k}^{(w)} = I_{t_k} \big|_{\Omega_w}.$$

Esta restricción es metodológicamente importante. Lo que interesa no es clasificar cualquier cobertura terrestre del entorno, sino caracterizar los materiales flotantes presentes sobre la superficie activa del embalse. Esa decisión reduce ambigüedad, mejora pertinencia ecológica y hace comparables los resultados entre fechas.

La justificación institucional del desarrollo subraya precisamente que el proceso fue concebido para recolectar datos de un área específica correspondiente al espejo de agua y, a partir de allí, clasificar la superficie en cuatro categorías y estimar el área ocupada por cada una en km^2 y en porcentaje respecto del total evaluado [2, 4]. El objetivo no es solamente producir una cartografía temática, sino construir indicadores de ocupación que permitan seguir tendencias y cambios en el tiempo.

9.3. Contexto ambiental y motivación de monitoreo

La propuesta de investigación vinculada a este desarrollo ubica el problema en el embalse del Proyecto Hidroeléctrico Ituango, donde la acumulación de materiales flotantes ha estado asociada a arrastre de biomasa leñosa, propagación de macrófitas acuáticas, especialmente *Eichhornia crassipes*, y ocasional aparición de fitoplancton y cianobacterias [6]. Estas dinámicas no solo afectan la lectura ecológica del sistema. También inciden en la operación, la navegabilidad, el seguimiento ambiental y la definición de medidas de

manejo.

La maestría plantea, además, que la variación espacial y temporal de estos materiales está influida por variables climatológicas y de calidad del agua, y que por ello el análisis de imágenes satelitales debe dialogar con mediciones *in situ* y con una comprensión más amplia del comportamiento ambiental del embalse [6]. Este punto es importante para el capítulo: la clasificación aquí descrita no pretende agotar todo el problema, pero sí construir una base cuantitativa confiable sobre la que luego puedan articularse análisis ecológicos y correlaciones con variables fisicoquímicas.

Desde esta perspectiva, el trabajo no se reduce a identificar manchas de color en una imagen. Busca construir una representación operativa de la dinámica superficial del embalse. La clasificación se vuelve así una mediación entre la imagen satelital y la gestión ambiental.

Band	Name	Wavelength (fwhm)	Interoperable with Sentinel-2
1	Coastal Blue	443 (20)	Yes - with Sentinel-2 band 1
2	Blue	490 (50)	Yes - with Sentinel-2 band 2
3	Green I	531 (36)	No equivalent with Sentinel-2
4	Green	565 (36)	Yes - with Sentinel-2 band 3
5	Yellow	610 (20)	No equivalent with Sentinel-2
6	Red	665 (31)	Yes - with Sentinel-2 band 4
7	Red Edge	705 (15)	Yes - with Sentinel-2 band 5
8	NIR	865 (40)	Yes - with Sentinel-2 band 8a

Figura 9.2: Esquema conceptual de los datos de entrada PlanetScope, destacando las ocho bandas utilizadas en la diferenciación de coberturas sobre el espejo de agua.

9.4. Arquitectura general del proceso

La lógica completa del desarrollo puede resumirse como una cadena de cuatro etapas:

$$\mathcal{S} = \mathcal{Q} \circ \mathcal{C} \circ \mathcal{V} \circ \mathcal{L},$$

donde \mathcal{L} representa la carga y ensamblaje de las bandas de entrada, \mathcal{V} la visualización y exploración de la imagen, \mathcal{C} la clasificación de la superficie y \mathcal{Q} la cuantificación y exportación de resultados.

La primera etapa no consiste simplemente en abrir un archivo. El manual de usuario indica

que el usuario selecciona el archivo `stk.adf`, correspondiente a una imagen previamente procesada, y el sistema se encarga de combinar las ocho bandas originales en una sola imagen TIF multibanda [1]. La segunda etapa permite explorar la escena, verificar la representación RGB y navegar mediante herramientas de zoom y desplazamiento. La tercera activa el modelo de clasificación. La cuarta traduce esa clasificación en superficies y porcentajes, y habilita el guardado de una salida TIF y un archivo ASCII con el resumen estadístico [1].

Si se denota por $X_n \in \mathbb{R}^8$ el vector espectral del píxel n perteneciente al dominio activo Ω_w , el problema de clasificación puede expresarse como la estimación de una función

$$f : \mathbb{R}^8 \rightarrow \mathcal{K},$$

donde

$$\mathcal{K} = \{\text{Water, Wood, Sediments, Aquatic vegetation}\}.$$

Así, para cada píxel activo se obtiene

$$\hat{y}_n = f(X_n),$$

siendo \hat{y}_n la clase asignada.

Una vez clasificados todos los píxeles, la salida puede representarse como un mapa temático

$$C(x, y) \in \mathcal{K}, \quad (x, y) \in \Omega_w.$$

Ese mapa ya no contiene únicamente valores radiométricos o espectrales, sino una interpretación semántica del espacio superficial del embalse. La cuantificación posterior consiste en contar la extensión de cada clase sobre dicho dominio.

9.5. Datos de entrada y representación espacial

Tanto el manual técnico como el manual de usuario señalan que las imágenes de entrada son rásteres con extensión `*.adf`, ocho bandas, tamaño de píxel de 3×3 metros, valores enteros sin signo y profundidad de 32 bits, georreferenciados en el sistema WGS_1984_UTM_Zone_18N [1, 2]. También se indica que para calibración y entrenamiento del sistema se emplearon 160 imágenes con cubrimiento temporal de los años 2020, 2021 y 2022 y un cubrimiento espacial promedio de 3244 ha, variable según la cota de inundación [1, 2].

Si cada píxel tiene lado $r = 3$ m, entonces el área elemental por píxel es

$$a_p = r^2 = 9 \text{ m}^2.$$

Expresado en kilómetros cuadrados,

$$a_p^{(km^2)} = \frac{9}{10^6} = 9 \times 10^{-6} \text{ km}^2.$$

Esta conversión es la base geométrica que permite transformar conteos de píxeles clasificados en estimaciones de superficie por categoría.

Si N_k es el número de píxeles asignados a la clase k , entonces el área ocupada por dicha categoría es

$$A_k = N_k \cdot a_p^{(km^2)}.$$

Por su parte, si N_Ω es el número total de píxeles del dominio activo del espejo de agua, el porcentaje de ocupación de la clase k se calcula como

$$P_k = \frac{N_k}{N_\Omega} \times 100.$$

Estas expresiones condensan el núcleo cuantitativo del proceso. La clasificación tiene valor ambiental precisamente porque no termina en una imagen coloreada. Se convierte en área y porcentaje, es decir, en magnitudes comparables entre fechas y útiles para seguimiento.

9.6. Interfaz y secuencia operativa

Una virtud importante del desarrollo es que traduce una cadena técnica relativamente compleja a una interfaz clara para el usuario. El manual de usuario muestra una ventana principal con un área de visualización a la izquierda y un panel de control a la derecha, donde se ubican los botones de carga, clasificación y guardado, así como un panel informativo con los valores de área y porcentaje para cada clase [1]. Esa organización refleja bien la intención del proceso: primero observar, luego clasificar, finalmente cuantificar y guardar.

La secuencia de uso puede representarse como una máquina de estados:

$$s_0 \xrightarrow{\text{Upload file}} s_1 \xrightarrow{\text{Classification}} s_2 \xrightarrow{\text{Save}} s_3,$$

donde

- s_0 : sin imagen cargada,
- s_1 : imagen multibanda cargada y visualizable,
- s_2 : imagen clasificada con métricas desplegadas,
- s_3 : resultados exportados.

El manual también documenta funciones auxiliares para exploración visual, como *Original image*, *Zoom In* y *Zoom Out*, además de barras de desplazamiento para navegación en la imagen [1]. Aunque estas herramientas puedan parecer secundarias, cumplen una función metodológica importante: permiten al usuario verificar zonas de interés, inspeccionar detalles y contrastar visualmente la salida del proceso con la representación RGB de entrada.

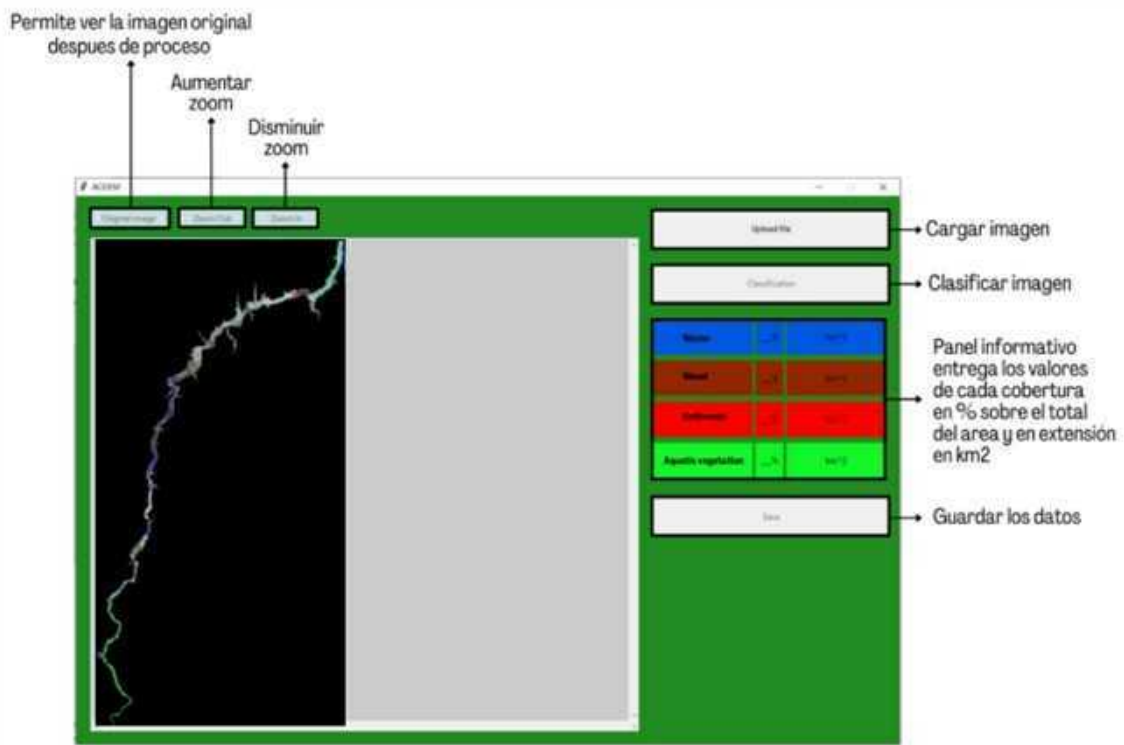


Figura 9.3: Interfaz principal del sistema, con área de visualización, panel lateral de resultados y secuencia de carga, clasificación y guardado.

9.7. Clasificación de la superficie y construcción de la salida temática

Una vez cargada la imagen, el usuario activa el botón *Classification*. El manual de usuario indica que durante esta etapa el sistema ejecuta el proceso de clasificación, muestra una señal visual de que el procedimiento está en curso y, al finalizar, llena los espacios del panel lateral con el porcentaje y el área correspondiente a cada categoría [1]. El manual técnico complementa esta descripción al señalar que la función `clasif_img` llama a `aplicar_modelo`, la cual carga la imagen de ocho bandas desde la carpeta temporal, aplica el modelo de clasificación, crea una nueva imagen y genera un archivo de texto con el aforo de áreas [2].

Desde una perspectiva matemática, si el mapa clasificado final se denota por $C(x, y)$, puede interpretarse como una variable categórica definida sobre el dominio activo:

$$C : \Omega_w \rightarrow \mathcal{K}.$$

Para efectos computacionales, esa variable puede codificarse también mediante cuatro bandas binarias:

$$B_k(x, y) = \begin{cases} 1, & \text{si } C(x, y) = k, \\ 0, & \text{en otro caso.} \end{cases}$$

Así, la imagen clasificada exportada en formato TIF puede representarse como el conjunto

$$\mathcal{B} = \{B_1, B_2, B_3, B_4\},$$

donde cada banda corresponde a una clase: agua, maderos flotantes, sedimentos y vegetación acuática. Esta descripción coincide con lo documentado en el manual de usuario, donde se señala que la salida TIF contiene cuatro bandas, una por cada categoría [1].

Esta decisión de diseño es metodológicamente valiosa. Al conservar las clases en bandas separadas, la salida no solo sirve para visualización inmediata, sino también para uso posterior en entornos SIG como QGIS o ArcMap, donde puede ser reutilizada para cartografía, análisis espacial detallado o generación de productos derivados [1].

9.8. Cuantificación y exportación de resultados

La clasificación por sí sola produce una interpretación espacial. Pero el objetivo de monitoreo exige ir más allá: cuantificar. La cuantificación es lo que permite comparar fechas,

analizar tendencias y comunicar resultados de forma operativa. El manual de usuario especifica que, una vez finalizado el proceso, el sistema muestra para cada clase su porcentaje respecto al área activa y el área correspondiente en km^2 , y que el botón *Save* guarda tanto la imagen clasificada como un archivo de texto plano con el resumen estadístico [1].

Si $k \in \mathcal{K}$ es una de las clases de interés, entonces los dos indicadores principales del sistema son:

$$A_k = N_k \cdot a_p^{(\text{km}^2)}$$

y

$$P_k = \frac{N_k}{N_\Omega} \times 100.$$

El primero expresa extensión absoluta. El segundo expresa ocupación relativa. Ambos son necesarios. Un crecimiento en porcentaje no siempre implica el mismo impacto absoluto si la cota del embalse varía; del mismo modo, un área absoluta mayor puede ser más o menos relevante según la superficie disponible en la fecha observada. La combinación de ambos indicadores en la interfaz resulta adecuada para un monitoreo comparativo más sólido.

La estructura de salida documentada en el manual incluye campos como identificador de objeto, banda, clasificación, porcentaje y área, lo que convierte la exportación en una tabla simple pero suficientemente informativa para análisis posteriores [1]. Esta salida puede verse como la tabla

$$\mathcal{T} = \{(k, A_k, P_k)\}_{k=1}^4,$$

es decir, un resumen por clase de la ocupación de la superficie evaluada.



Figura 9.4: Ejemplo de salida clasificada, con diferenciación de las cuatro categorías y resumen de áreas y porcentajes por clase.

9.9. Arquitectura técnica del desarrollo

El manual técnico permite reconstruir con claridad la estructura del sistema. Se documentan librerías principales como `numpy`, `tkinter`, `cv2`, `sklearn`, `rasterio`, `pandas`, `pickle`, `PIL` y `pathlib` [2]. También se describen funciones principales como `askfile`, `comb_bandas`, `clasif_img`, `aplicar_modelo`, `save_data`, `original`, `zoomOut` y `zoomIn` [2].

Una forma compacta de representar la arquitectura del sistema es mediante la tupla

$$\mathcal{A} = (\mathcal{G}, \mathcal{R}, \mathcal{M}, \mathcal{E}),$$

donde:

- \mathcal{G} = módulo gráfico e interacción con el usuario,
- \mathcal{R} = módulo de lectura y combinación de bandas ráster,
- \mathcal{M} = módulo de clasificación y aforo,
- \mathcal{E} = módulo de exportación de resultados.

En términos operativos, el flujo principal del código puede resumirse como

```
save_data ◦ clasif_img ◦ comb_bandas ◦ askfile.
```

La secuencia expresa lo que realmente ocurre: primero se localiza el archivo de entrada, luego se ensamblan las ocho bandas, después se aplica el clasificador y finalmente se exportan los productos.

También es relevante señalar que el manual técnico informa que el sistema fue desarrollado en Windows 10, usando Spyder sobre Anaconda con Python 3.11.5, y que se ensayó en equipos con procesador AMD Ryzen 2200G, 8 GB de RAM, 4 GB de espacio disponible y GPU Radeon Vega 8 [2]. Esto sitúa el desarrollo en un entorno de cómputo accesible para laboratorios universitarios y refuerza su carácter de capacidad instalada real, no meramente conceptual.

9.10. Relación con la investigación de maestría y alcance metodológico

Uno de los aspectos más valiosos de este trabajo es que el desarrollo computacional no aparece desconectado de una agenda científica, sino articulado a una investigación de maestría. La propuesta de Ana María Marín Guerra plantea explícitamente el análisis espacio-temporal de la dinámica de la cobertura de materiales flotantes en el embalse de Hidroituango y sugiere una metodología que combina correcciones radiométricas y atmosféricas, clasificación por píxel, índices de vegetación, técnicas de visión artificial y correlación con variables ambientales [6].

Dentro de esa agenda, el proceso de clasificación aquí descrito ocupa un lugar central. En primer lugar, porque permite generar una base cuantitativa sobre la distribución espacial de las coberturas flotantes. En segundo lugar, porque habilita comparaciones entre momentos de observación. Y en tercer lugar, porque constituye una entrada necesaria para análisis posteriores sobre biomasa, tasas de crecimiento, variación estacional y relación con condiciones climatológicas o fisicoquímicas.

Si se define

$$A_k(t)$$

como el área ocupada por la clase k en el instante t , entonces el seguimiento multitemporal de una cobertura puede representarse como la serie

$$\{A_k(t_1), A_k(t_2), \dots, A_k(t_T)\}.$$

De manera análoga, su comportamiento relativo puede estudiarse mediante

$$\{P_k(t_1), P_k(t_2), \dots, P_k(t_T)\}.$$

Estas series constituyen la base para analizar tendencias, tasas de cambio o episodios de expansión y contracción de materiales flotantes.

La certificación de uso refuerza esta lectura, al indicar que el desarrollo está siendo utilizado tanto en el proyecto SIMEVAM como en una maestría [5]. Esto confirma que el valor del proceso no reside solamente en clasificar una imagen aislada, sino en integrarse a un programa de investigación más amplio sobre monitoreo ambiental.

9.11. Protección intelectual, uso y fortalecimiento de capacidades

El registro de la DNDA describe la obra como un software que permite al usuario ejecutar un procesamiento sobre imágenes para recolectar datos del espejo de agua del embalse y clasificar la superficie en cuatro categorías, entregando área y porcentaje por clase [4]. El registro identifica como autores a David Fernández Mc Cann, Fabio Vélez Macías, Juan Guillermo Quevedo y Ana María Marín Guerra, y establece a la Universidad de Antioquia como titular patrimonial y productor [4].

La solicitud institucional agrega un aspecto especialmente importante para el seminario: el desarrollo fortaleció capacidades del grupo mediante formación de estudiantes de pregrado y maestría, adquisición de conocimientos especializados en visión artificial y consolidación de recursos reutilizables como software y manuales [3]. Además, se afirma que el grupo quedó en capacidad de ofrecer servicios avanzados de evaluación de coberturas flotantes basados en este proceso [3].

Todo esto permite afirmar que el aporte del trabajo no se agota en su función inmediata. También dejó infraestructura metodológica, capital humano entrenado y una plataforma concreta para futuras investigaciones en el ámbito del monitoreo de embalses y cuerpos de agua.

9.12. Discusión

Este capítulo no se centra en el nombre del software, porque la contribución de fondo no es nominal, sino metodológica. Lo verdaderamente importante aquí es la construcción de

un proceso reproducible para transformar imágenes satelitales multiespectrales en información ambiental cuantificable sobre materiales flotantes. Ese proceso tiene valor porque toma un problema real de monitoreo y lo organiza en etapas claras: selección del dominio activo, clasificación de la superficie, cálculo de áreas, cálculo de porcentajes y exportación de resultados.

También conviene subrayar que la metodología no pretende resolver por sí sola toda la complejidad ecológica del embalse. La propuesta de maestría muestra que todavía existe un horizonte más amplio: correlacionar estas salidas con biomasa, variables fisicoquímicas, clima y dinámica de especies como *Eichhornia crassipes* [6]. Pero precisamente ahí está su fortaleza. La clasificación constituye una base sólida, no un cierre artificial del problema.

Desde la perspectiva del *Seminario GeoLimna–GEPAR*, el trabajo ocupa un lugar especialmente sugestivo porque articula percepción remota, procesamiento digital de imágenes, visión artificial, interfaz utilizable, protección intelectual y uso institucional. Ese ensamblaje es el que convierte un desarrollo de software en una capacidad científica y tecnológica madura.

9.13. Conclusiones

Este capítulo presentó una metodología para la caracterización multitemporal de coberturas flotantes en embalses mediante procesamiento de imágenes satelitales multiespectrales. Se mostró que el proceso parte de imágenes PlanetScope previamente corregidas y recortadas al espejo de agua, clasifica la superficie en cuatro categorías operativamente relevantes y entrega como resultado tanto una salida cartográfica como indicadores cuantitativos de área y porcentaje por clase.

La formulación matemática permitió expresar con claridad la transición desde el píxel multiespectral hacia la clase temática, y desde el conteo de píxeles hacia métricas espaciales comparables entre fechas. La descripción de la interfaz y de la arquitectura técnica mostró, además, que la metodología fue encapsulada en una herramienta funcional, documentada y reutilizable.

Su valor principal, sin embargo, no está en el nombre de la aplicación que la implementa, sino en el proceso que deja establecido. Ese proceso convierte una necesidad de monitoreo ambiental en una secuencia verificable de tratamiento de datos, clasificación y cuantificación. En ese sentido, constituye una base metodológica sólida para el estudio de la dinámica de materiales flotantes, para la gestión ambiental de embalses y para investigaciones futuras que integren teledetección, variables ambientales y análisis ecológico multitemporal.

Agradecimientos

Se agradece a la Universidad de Antioquia y a los grupos de investigación GeoLimna y GEPAR por el entorno académico, técnico e institucional que hizo posible este desarrollo. Se reconoce de manera especial la articulación del trabajo con el proyecto SIMEVAM y con el proceso de formación de maestría asociado al análisis espacio-temporal de coberturas flotantes en el embalse de Hidroituango [5, 6]. Asimismo, se reconoce la contribución de los autores registrados ante la DNDA y el valor del desarrollo como capacidad instalada para futuras investigaciones y servicios tecnológicos [3, 4].

Referencias

- [1] *ACOEM: Caracterización multitemporal de coberturas flotantes a partir del procesamiento de imágenes satelitales de alta resolución espacial y espectral. Manual de usuario*, Universidad de Antioquia, Medellín, Colombia, 2023.
- [2] *ACOEM: Caracterización multitemporal de coberturas flotantes a partir del procesamiento de imágenes satelitales de alta resolución espacial y espectral. Manual técnico*, Universidad de Antioquia, Medellín, Colombia, 2023.
- [3] D. S. Fernández Mc Cann, F. d. J. Vélez Macías et al., *Solicitud para asignación de software ACOEM*, Documento institucional, Universidad de Antioquia, Medellín, Colombia, 2024.
- [4] Dirección Nacional de Derecho de Autor, *ACOEM: Caracterización temporal de suelos usando imágenes digitales. Certificado de registro de soporte lógico–software*, Libro 13-97-386, fecha de registro: 29-ene-2024, 2024.
- [5] F. d. J. Vélez Macías y D. S. Fernández Mc Cann, *Certificación de uso del software ACOEM*, Certificación institucional, Universidad de Antioquia, Medellín, Colombia, jun. de 2024.
- [6] A. M. Marín Guerra, *Análisis espacio temporal de la dinámica de la cobertura de los materiales flotantes en el embalse de Hidroituango*, Propuesta de Maestría en Ingeniería Ambiental, Universidad de Antioquia, 2024.
- [7] K. H. Thamaga y T. Dube, «Remote sensing of invasive water hyacinth (*Eichhornia crassipes*): A review on applications and challenges,» *Remote Sensing Applications: Society and Environment*, vol. 10, págs. 36-46, 2018. DOI: 10.1016/j.rsase.2018.02.005.

- [8] L. Pádua, A. M. Antão-Geraldes, J. J. Sousa et al., «Water Hyacinth (*Eichhornia crassipes*) Detection Using Coarse and High Resolution Multispectral Data,» *Drones*, 2022. DOI: 10.3390/drones6020047.

Capítulo 10

Identificación asistida de tiburones mediante análisis de aletas dorsales y descriptores de Fourier

Autores

Mariana Paniagua Lopera¹; Juan Pablo Serna López²; Karla Cirila Garcés García³; David Stephen Fernández Mc Cann⁴.

Resumen

La identificación taxonómica rápida y confiable de tiburones en puntos de desembarque, campañas de monitoreo y contextos de investigación marina sigue siendo un reto técnico y operativo. Las dinámicas propias de la pesca artesanal y comercial, la necesidad de decisiones ágiles en campo y la dependencia de personal altamente especializado hacen que los registros de captura no siempre alcancen el nivel de precisión deseado. En este capítulo se presenta un proceso de identificación asistida de tiburones a partir del análisis de la aleta dorsal, integrando procesamiento digital de imágenes, segmentación basada en inteligencia artificial y segmentación clásica por color, con posterior extracción de descriptores genéricos de Fourier y un mecanismo de decisión por similitud. El procedimiento

¹Estudiante de Ingeniería Electrónica.

²Profesor e investigador, Universidad de Antioquia, Seccional Urabá.

³Profesora e investigadora, Universidad Veracruzana.

⁴Profesor, Departamento de Ingeniería Electrónica y Telecomunicaciones.

^{01,2,4}Universidad de Antioquia. Grupos de investigación GeoLimna y GEPAR.

⁰³Universidad Veracruzana.

fue planteado para seis especies comunes en aguas de México y Colombia: *Carcharhinus falciformis*, *Galeocerdo cuvier*, *Carcharhinus limbatus*, *Carcharhinus obscurus*, *Carcharhinus perezii* y *Sphyrna lewini*. La metodología organiza una cadena de etapas que va desde la adquisición de imágenes y la selección del área de interés hasta la obtención de una recomendación de especie basada en la forma de la aleta. Más allá de la herramienta computacional que la implementa, la contribución central del trabajo está en estructurar un proceso reproducible, interpretable y suficientemente ligero para facilitar el monitoreo de tiburones en escenarios reales de campo.

Palabras clave: visión artificial; tiburones; aleta dorsal; descriptores de Fourier; segmentación; monitoreo pesquero; conservación marina.

10.1. Introducción

La identificación de tiburones en puntos de desembarque pesquero es una actividad crucial para la gestión del recurso, pero suele desarrollarse bajo condiciones poco favorables para la precisión taxonómica. El tiempo disponible es reducido, el estado físico de los individuos puede variar, las condiciones de iluminación no siempre son ideales y, en muchos casos, la identificación depende del criterio experto de biólogos o técnicos con amplia experiencia. Esta situación se vuelve todavía más crítica en países en desarrollo, donde el monitoreo por especie no siempre se realiza de forma sistemática y donde la presión operativa de la actividad pesquera puede superar la capacidad de registro detallado.

En el contexto colombiano y mexicano, el problema tiene un peso particular. La documentación asociada al proyecto recuerda que México cuenta con 111 especies de tiburones y Colombia con 76, lo que ubica a ambos países entre los de mayor diversidad de peces cartilagosos en América Latina. Al mismo tiempo, los tiburones tienen un valor ecológico y, en algunos contextos, también económico, por lo que disponer de herramientas de identificación más ágiles resulta fundamental para monitoreo, manejo y conservación. En Colombia, además, la captura de tiburones se encuentra fuertemente regulada, lo que da aún más importancia a una determinación correcta de las especies presentes en campo.

La propuesta que aquí se desarrolla parte de una idea concreta: la forma de la aleta dorsal contiene información morfológica suficiente para apoyar la discriminación entre especies, siempre que pueda ser capturada, segmentada y descrita de manera consistente. Desde esa premisa, el proceso se estructura en torno a cuatro elementos: adquisición de imágenes, selección del área de interés, segmentación de la aleta y descripción matemática de su contorno mediante descriptores de Fourier. El resultado no es solamente una clasificación computacional, sino una forma de traducir la forma biológica en un vector de

características comparable entre individuos.

Este trabajo también tiene un trasfondo institucional e internacional que conviene resaltar. La propuesta se articula entre la Universidad de Antioquia y la Universidad Veracruzana, y su desarrollo fue presentado en el *9th World Fisheries Congress* en Seattle, bajo el título *A novel application based on the Fourier coefficients to identify sharks*. Esa proyección externa confirma que el proceso no se limita a una exploración académica local, sino que dialoga con preocupaciones globales sobre trazabilidad, monitoreo y sostenibilidad pesquera.



Figura 10.1: Escena representativa de muestreo o desembarque, correspondiente al contexto real en el que se requiere una identificación rápida y precisa de tiburones.

10.2. Problema y motivación del proceso

El problema puede formularse de la siguiente manera: dada una imagen de un tiburón completo o de su aleta dorsal, se requiere un procedimiento que permita inferir a cuál de un conjunto reducido de especies pertenece el individuo, utilizando rasgos morfológicos extraídos de la forma de la aleta y evitando, en lo posible, la dependencia exclusiva de inspección manual experta.

Sea

$$I : \Omega \subset \mathbb{R}^2 \rightarrow \mathbb{R}^3$$

una imagen en color, donde Ω representa el dominio espacial y el codominio corresponde a la representación cromática del píxel. El objetivo del proceso no es analizar toda la escena indiscriminadamente, sino localizar primero una región de interés que contenga la aleta dorsal. Si la región seleccionada por el usuario está centrada en una ventana $\Omega_f \subset \Omega$, entonces la imagen de trabajo queda definida por

$$I_f = I|_{\Omega_f}.$$

Esta restricción es importante porque la aleta dorsal es el soporte morfológico principal de la metodología. Tanto la propuesta de grado como los manuales de uso y técnicos dejan claro que el flujo operativo exige seleccionar la zona de la imagen donde se encuentra la aleta antes de proceder con la identificación. En otras palabras, el sistema no reemplaza completamente al observador humano, sino que se apoya en él para ubicar el objeto biológico relevante, y a partir de allí automatiza la parte más analítica del proceso.

La motivación es doble. En primer lugar, existe una necesidad práctica de mejorar la velocidad y la precisión de la identificación en condiciones de campo. En segundo lugar, existe una necesidad científica de transformar esas identificaciones en registros más consistentes para inventarios, monitoreo poblacional y estrategias de conservación. El proceso propuesto intenta responder a ambas necesidades mediante una arquitectura suficientemente ligera para operar con recursos moderados, pero lo bastante rigurosa para apoyarse en descriptores geométricos y criterios explícitos de comparación.

10.3. Especies objetivo y criterio morfológico de discriminación

La metodología fue planteada para seis especies de tiburón comunes en los mares colombianos y mexicanos y relevantes desde el punto de vista ecológico o pesquero:

$$\mathcal{S} = \left\{ \begin{array}{l} Carcharhinus falciformis, Galeocerdo cuvier, Carcharhinus limbatus, \\ Carcharhinus obscurus, Carcharhinus perezii, Sphyrna lewini \end{array} \right\}.$$

Aunque el sistema completo puede evolucionar hacia más partes del cuerpo o más especies, el núcleo del proceso descrito en los documentos y resultados se centra en la aleta dorsal. Esta decisión tiene una lógica biológica y computacional. Desde el punto de vista biológico, la aleta dorsal presenta diferencias de tamaño relativo, curvatura, inclinación y geometría del borde que suelen ser útiles para el reconocimiento por parte de especialistas. Desde el punto de vista computacional, es una estructura cuya forma puede representarse por contorno y, por tanto, describirse de manera compacta mediante herramientas espectrales como los descriptores de Fourier.

Si C representa el contorno cerrado de la aleta ya segmentada, entonces el problema de identificación se reduce a encontrar una aplicación

$$g : C \rightarrow \mathbb{R}^m$$

que convierta la forma en un vector de características suficientemente estable y discriminante. En este trabajo, esa aplicación se implementa mediante descriptores genéricos de Fourier, y el espacio \mathbb{R}^m corresponde al conjunto de coeficientes seleccionados para representar la geometría de la aleta.

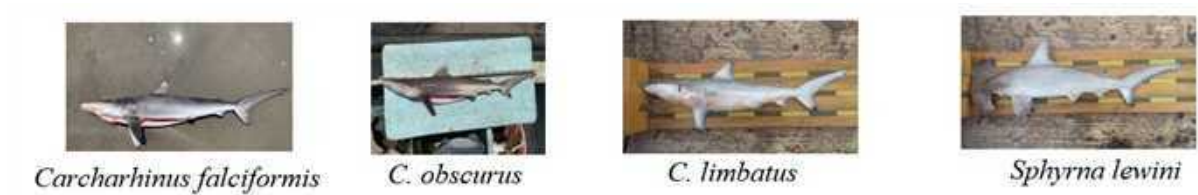


Figura 10.2: Ejemplos de especies objetivo y variabilidad morfológica de sus aletas dorsales.

10.4. Arquitectura general del proceso

La lógica completa del procedimiento puede resumirse como una cadena de cinco etapas:

$$\mathcal{P} = \mathcal{D} \circ \mathcal{R} \circ \mathcal{M} \circ \mathcal{F} \circ \mathcal{I},$$

donde \mathcal{I} corresponde a la adquisición de la imagen, \mathcal{F} a la selección de la región de interés, \mathcal{M} a la segmentación de la aleta, \mathcal{R} al cálculo de los descriptores de Fourier y \mathcal{D} al mecanismo de decisión o similitud entre la imagen de entrada y las referencias almacenadas.

La adquisición de la imagen puede provenir de dos fuentes principales: fotografías tomadas directamente en campo, especialmente en Tamiahua, Veracruz, por estudiantes y docentes vinculados a la Universidad Veracruzana, y fotografías seleccionadas desde plataformas colaborativas como iNaturalist. Una vez cargada la imagen, el usuario delimita el área donde se encuentra la aleta dorsal. Esa región seleccionada pasa luego a uno de dos esquemas de segmentación posibles: segmentación asistida por SAM o segmentación clásica por umbralización. Finalmente, sobre la máscara binaria obtenida se calculan los descriptores de Fourier y se comparan con descriptores de referencia por especie.

Matemáticamente, si I_f es la imagen recortada y M la máscara binaria correspondiente a la aleta, entonces se construye un vector descriptor

$$\mathbf{d} = [d_1, d_2, \dots, d_m]^\top,$$

donde cada d_i representa uno de los coeficientes seleccionados. El objetivo final del proceso

es decidir

$$\hat{s} = \arg \max_{s \in \mathcal{S}} \rho(\mathbf{d}, \bar{\mathbf{d}}_s),$$

donde $\bar{\mathbf{d}}_s$ es el descriptor promedio asociado a la especie s , y ρ representa una medida de similitud, en este caso una correlación o un criterio equivalente de cercanía.

Esta forma de escribir el problema deja claro que el sistema no opera directamente sobre intensidades crudas ni sobre etiquetas abstractas, sino sobre una representación intermedia de forma. El paso clave es, entonces, conseguir una máscara suficientemente fiel de la aleta.

10.5. Adquisición de datos y construcción de la base de referencia

La base de imágenes fue construida a partir de dos fuentes complementarias. La primera corresponde a fotografías tomadas en Tamiahua, Veracruz, México, por estudiantes y profesores vinculados a biología marina. La segunda fuente está formada por imágenes obtenidas de iNaturalist, una plataforma colaborativa de observación biológica. La lógica de combinar ambas fuentes fue sencilla: las imágenes de campo aportan realismo y relevancia ecológica, mientras que las imágenes de plataformas abiertas amplían la variabilidad visual disponible para cada especie.

El conjunto utilizado en la fase reportada del trabajo fue relativamente reducido. Los documentos del proyecto señalan que se trabajó con entre 15 y 30 imágenes por especie, mientras que el material presentado en el World Fisheries Congress habla de entre 16 y 25 imágenes procesadas por especie. Lejos de ser una debilidad trivial, este hecho es metodológicamente importante: muestra que el proceso fue diseñado para funcionar con bases de datos moderadas, sin exigir un entrenamiento masivo como el que requieren muchos sistemas de aprendizaje profundo de clasificación supervisada.

Si se define

$$\mathcal{D}_s = \{I_{s,1}, I_{s,2}, \dots, I_{s,n_s}\}$$

como el conjunto de imágenes disponibles para la especie s , entonces la base completa queda expresada como

$$\mathcal{D} = \bigcup_{s \in \mathcal{S}} \mathcal{D}_s.$$

A cada imagen de esta base se le asocia posteriormente una máscara de aleta y un vector descriptor. El paso de imagen a descriptor es el que permite construir una representación

promedio por especie:

$$\bar{\mathbf{d}}_s = \frac{1}{n_s} \sum_{j=1}^{n_s} \mathbf{d}_{s,j}.$$

Este promedio cumple una función fundamental: se convierte en la referencia contra la cual se compararán nuevas imágenes. En otras palabras, el sistema no aprende únicamente desde imágenes individuales, sino desde una representación agregada de la morfología dorsal de cada especie.

10.6. Selección del área de interés

Antes de segmentar, el proceso solicita una acción simple pero crucial del usuario: delimitar la región donde se encuentra la aleta dorsal. Tanto el manual de usuario como los materiales de apoyo muestran que, una vez cargada la imagen, aparece una ventana adicional en la cual el usuario dibuja un rectángulo sobre la zona de la aleta. El objetivo de esta etapa no es superficial. Al restringir el campo visual al objeto de interés, se reduce la probabilidad de que el sistema confunda otros elementos del entorno con la aleta y se mejora la precisión de la segmentación posterior.

Si las coordenadas del rectángulo seleccionado son (x_1, y_1) para la esquina superior izquierda y (x_2, y_2) para la esquina inferior derecha, entonces la región de interés queda definida por

$$\Omega_f = \{(x, y) \in \Omega : x_1 \leq x \leq x_2, y_1 \leq y \leq y_2\}.$$

La imagen de entrada al bloque de segmentación es entonces

$$I_f(x, y) = I(x, y), \quad (x, y) \in \Omega_f.$$

Esta formulación explica por qué el sistema puede operar de manera más robusta incluso cuando la imagen original incluye el cuerpo completo del tiburón, agua, otros organismos o elementos de fondo. El usuario orienta el proceso hacia la estructura anatómica relevante, y el algoritmo asume la tarea de separarla y describirla.



Figura 10.3: Selección de la región de interés sobre la aleta dorsal, como paso previo a la segmentación y al cálculo de descriptores.

10.7. Segmentación de la aleta dorsal

La metodología contempla dos rutas de segmentación. La primera utiliza el modelo *Segment Anything Model* (SAM), una herramienta de segmentación de instancias basada en inteligencia artificial. La segunda utiliza un procedimiento clásico por umbralización sobre fondo amarillo, acompañado de operaciones morfológicas de limpieza. La existencia de estas dos rutas es una decisión de ingeniería que permite adaptar el proceso al tipo de imagen disponible.

10.7.1. Segmentación con SAM

La ruta basada en SAM está pensada para imágenes más complejas, como aquellas donde aparece el tiburón completo, donde la aleta no está completamente separada del torso o donde el fondo no ha sido controlado experimentalmente. En este caso, el procedimiento parte de una conversión de BGR a RGB, compatible con las librerías empleadas para la segmentación, y luego utiliza SAM para generar una máscara del objeto.

Si la máscara resultante se representa por

$$M_{SAM}(x, y) \in \{0, 1\},$$

entonces

$$M_{SAM}(x, y) = 1$$

para los píxeles pertenecientes a la aleta y

$$M_{SAM}(x, y) = 0$$

para el fondo o el resto de la imagen. Esta máscara binaria permite extraer el contorno de la aleta y preparar la forma para el cálculo de descriptores.

El interés de esta ruta es claro: SAM genera bordes más suaves, define mejor la forma y elimina ruido de manera más eficaz que los métodos clásicos cuando la escena es compleja. Por ello, aunque exige mayor capacidad computacional, suele producir segmentaciones más limpias y, en consecuencia, descriptores más estables.

10.7.2. Segmentación clásica por umbralización

La segunda ruta fue diseñada para imágenes más controladas, especialmente aquellas en las que la aleta ya ha sido separada del cuerpo y colocada sobre un fondo amarillo. En este caso, el procedimiento transforma la imagen de BGR a HSV y luego aplica umbralización sobre un rango específico del color amarillo. El material metodológico reporta un rango inferior

$$[20, 100, 100]$$

y un rango superior

$$[30, 255, 255].$$

A partir de esos umbrales se construye una máscara inicial, y posteriormente se aplican erosiones para remover manchas y residuos del fondo.

Sea

$$M_{HSV}(x, y)$$

la máscara inicial obtenida por umbralización. Tras aplicar una erosión morfológica E , se obtiene una versión refinada

$$M_{cl}(x, y) = E(M_{HSV}(x, y)).$$

En términos prácticos, esta operación busca conservar la estructura principal de la aleta mientras elimina componentes pequeñas o ruido periférico.

La ventaja principal de esta ruta es su velocidad. Al no depender de un modelo de segmentación profunda, el costo computacional es mucho menor. La desventaja es que requiere

condiciones de captura más estrictas: fondo uniforme, aleta centrada, sin tocar bordes y sin objetos ajenos en la escena.



Figura 10.4: Comparación entre la segmentación mediante umbralización y morfología y la segmentación obtenida con SAM sobre imágenes de campo.

10.8. Descripción matemática de la forma mediante descriptores de Fourier

Una vez obtenida una máscara binaria suficientemente limpia, el paso siguiente consiste en describir la forma de la aleta mediante descriptores genéricos de Fourier. La motivación para emplear esta familia de descriptores es que la forma de la aleta no se representa simplemente como una imagen, sino como una señal geométrica que puede analizarse en términos de sus componentes frecuenciales.

Sea $M(x, y)$ la máscara binaria final de la aleta. A partir de ella se define una región y se identifica su contorno C . Si ese contorno se expresa en coordenadas polares respecto de un centro de masa (x_c, y_c) , entonces cada punto puede representarse por una distancia radial $r(\theta)$, donde θ es el ángulo respecto del centro. El objetivo es convertir esa representación geométrica en una secuencia susceptible de ser analizada por Fourier.

En términos generales, si $f(r, \theta)$ representa la función asociada a la región polar de la aleta, entonces su expansión en Fourier permite obtener coeficientes complejos

$$F_{mn}$$

que contienen información sobre la estructura radial y angular de la forma. Los descriptores genéricos de Fourier se construyen a partir de estos coeficientes, normalizados de tal manera que resulten más estables frente a traslación, escala y, en parte, rotación.

En la implementación reportada por el trabajo de grado, se utilizaron los primeros 30 descriptores como representación suficiente de la forma de la aleta. Esta decisión no fue

arbitraria, sino producto de un análisis visual del comportamiento de los coeficientes, concluyéndose que esos primeros términos concentraban la mayor parte de la información significativa de la forma. Por ello, el vector descriptor de una imagen se expresa como

$$\mathbf{d} = [d_1, d_2, \dots, d_{30}]^\top.$$

A partir de estos vectores se calculó, para cada especie, un descriptor promedio:

$$\bar{\mathbf{d}}_s = \frac{1}{n_s} \sum_{j=1}^{n_s} \mathbf{d}_{s,j},$$

donde n_s es el número de imágenes disponibles para la especie s . Esta construcción es importante porque reduce la variabilidad entre individuos de una misma especie y genera una firma promedio que puede actuar como referencia en la fase de identificación.

10.9. Mecanismo de decisión por similitud

Con la imagen nueva ya segmentada y con su vector de 30 descriptores calculado, el proceso de identificación consiste en comparar ese vector con los descriptores promedio de las seis especies consideradas. El trabajo desarrollado emplea una estrategia de comparación basada en correlación, específicamente una correlación de Pearson, para medir la cercanía entre el vector de entrada y cada uno de los vectores de referencia.

Si \mathbf{d} es el vector descriptor de la imagen a identificar y $\bar{\mathbf{d}}_s$ el vector promedio de la especie s , entonces la medida de similitud puede expresarse como

$$\rho_s = \frac{\sum_{i=1}^{30} (d_i - \mu_d)(\bar{d}_{s,i} - \mu_s)}{\sqrt{\sum_{i=1}^{30} (d_i - \mu_d)^2} \sqrt{\sum_{i=1}^{30} (\bar{d}_{s,i} - \mu_s)^2}},$$

donde μ_d es la media de los descriptores de la imagen de entrada y μ_s la media de los descriptores promedio de la especie s .

La decisión final puede entonces escribirse como

$$\hat{s}_1, \hat{s}_2 = \text{Top2}\{\rho_s : s \in \mathcal{S}\},$$

es decir, el sistema no devuelve una sola especie, sino las dos especies con mayor similitud. Esta salida es metodológicamente sensata, porque reconoce que la forma de algunas aletas puede ser morfológicamente cercana entre especies y ofrece al usuario una recomendación

principal acompañada de una segunda alternativa razonable.

En términos de interfaz, este resultado se acompaña además de la imagen original, la aleta segmentada y el histograma o gráfica de los descriptores obtenidos, lo que convierte la salida en una forma de interpretación visual más rica que una etiqueta aislada.

10.10. Resultados y comportamiento del proceso

Los resultados reportados en el trabajo de grado muestran que el sistema fue evaluado con un conjunto de 40 imágenes nuevas, es decir, imágenes no usadas previamente en la construcción de los descriptores promedio. De ese conjunto, 28 fueron identificadas correctamente, lo cual corresponde a una precisión global de

$$\text{Acc} = \frac{28}{40} = 0,70.$$

Expresado en porcentaje:

$$\text{Acc} = 70 \%$$

Esta precisión debe interpretarse con cuidado. No representa un techo definitivo del proceso, sino el desempeño alcanzado bajo condiciones de una base de datos todavía limitada. Los mismos documentos enfatizan que se trabajó con un número reducido de imágenes por especie y que la diversidad visual del conjunto era todavía restringida. Bajo esas condiciones, obtener una precisión del 70 % es un resultado razonable y prometedor, especialmente considerando que la metodología no depende de una base masiva de entrenamiento ni de una red profunda para la clasificación final.

Además del valor agregado del 70 %, el material de resultados por especie sugiere precisiones diferenciadas. En una de las tablas preparadas para artículo se reportan, por ejemplo, valores de precisión de 68.75 % para *Galeocerdo cuvier*, 60 % para *Carcharhinus falciformis*, 75 % para *Sphyrna lewini*, 80 % para *Carcharhinus limbatus*, 66.67 % para *Carcharhinus obscurus* y 71.43 % para *Carcharhinus perezi*, con un promedio general de 70.31 %. Estas diferencias son coherentes con la variabilidad interespecífica de la forma de la aleta y con el tamaño desigual de los conjuntos de referencia.

También se observó que el método basado en SAM tiende a producir mejores segmentaciones y, por ende, descriptores más representativos, especialmente en imágenes complejas o con más elementos de fondo. El método clásico, por su parte, conserva valor cuando se dispone de condiciones de captura controladas y se requiere velocidad de procesamiento.

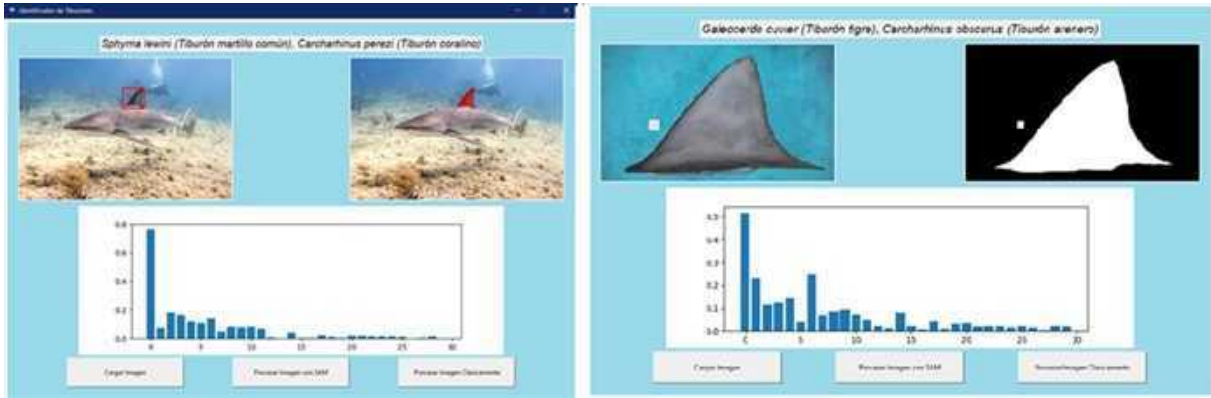


Figura 10.5: Resultado final del proceso, con la imagen original, la aleta segmentada, la gráfica de descriptores y las dos especies sugeridas por el sistema.

10.11. Arquitectura técnica e implementación

El manual técnico permite reconstruir con precisión la estructura funcional del sistema. El desarrollo fue realizado en Python 3.10.5, utilizando Visual Studio Code y entornos virtuales, con librerías como OpenCV, Tkinter, Pandas, NumPy, PIL, Scikit-image, Torch y el paquete asociado a SAM. Desde el punto de vista del hardware, el sistema puede funcionar en un equipo relativamente modesto para la ruta clásica, mientras que el uso de SAM se beneficia de una GPU dedicada.

La arquitectura funcional puede representarse por la tupla

$$\mathcal{A} = (\mathcal{U}, \mathcal{S}_{AI}, \mathcal{S}_{CL}, \mathcal{G}, \mathcal{C}),$$

donde \mathcal{U} es el módulo de interfaz y carga de imagen, \mathcal{S}_{AI} el módulo de segmentación con SAM, \mathcal{S}_{CL} el módulo de segmentación clásica, \mathcal{G} el cálculo de descriptores genéricos de Fourier y \mathcal{C} el módulo de correlación y decisión.

En la documentación del código aparecen funciones como `cargar`, `process_image`, `sam_img`, `seg_img`, `gfd` y `correlación`. Su secuencia funcional se puede resumir como

$$\text{correlación} \circ \text{gfd} \circ \begin{cases} \text{sam_img}, \\ \text{seg_img} \end{cases} \circ \text{process_image} \circ \text{cargar}.$$

Esta expresión resume bien la lógica del sistema: cargar la imagen, delimitar la región de la aleta, elegir una ruta de segmentación, calcular descriptores y comparar contra las referencias de especie.

10.12. Discusión

La contribución central de este trabajo no debe entenderse como la creación de una interfaz aislada, sino como la estructuración de un proceso de identificación que traduce un criterio morfológico de uso experto a una secuencia computacional reproducible. Ese matiz es importante. Mientras los métodos tradicionales dependen de observación experta y consulta de guías, aquí la forma de la aleta se convierte en un objeto matemático comparable.

También es relevante que el proceso combine dos filosofías de segmentación. Por un lado, aprovecha una herramienta de inteligencia artificial moderna como SAM, particularmente útil para escenas complejas. Por otro, conserva una ruta clásica por umbralización que resulta muy eficiente cuando la captura de la imagen puede controlarse. Esta coexistencia no debe verse como redundancia, sino como una forma de adaptar la tecnología a distintos escenarios de uso.

Desde la perspectiva del monitoreo pesquero y la conservación, el valor potencial del proceso es evidente. Una herramienta ligera, interpretable y relativamente precisa puede ayudar a mejorar los registros de captura, apoyar procesos de formación y reducir la dependencia absoluta de especialistas en situaciones donde el tiempo y los recursos son limitados. Además, la presentación del trabajo en el *World Fisheries Congress* sugiere que el enfoque tiene resonancia más allá del contexto local.

Las limitaciones también deben mencionarse con honestidad. El tamaño del conjunto de datos todavía es reducido, las imágenes pueden variar ampliamente en calidad y la discriminación entre especies morfológicamente cercanas sigue siendo un reto. Sin embargo, esas limitaciones no invalidan el enfoque; más bien muestran con claridad la ruta natural de crecimiento: ampliar la base de imágenes, robustecer la representación por especie y evaluar variantes más refinadas del esquema de similitud.

10.13. Conclusiones

Este capítulo presentó un proceso de identificación asistida de tiburones basado en la aleta dorsal como estructura morfológica principal, integrando selección de región de interés, segmentación mediante inteligencia artificial o umbralización clásica, extracción de descriptores genéricos de Fourier y un criterio de decisión por similitud con referencias de especie.

La importancia del trabajo no reside únicamente en haber implementado un programa

funcional, sino en haber formalizado una metodología interpretable que convierte la forma de la aleta en una representación numérica útil para apoyar la clasificación taxonómica en campo. La formulación matemática permitió mostrar con claridad cómo la imagen pasa a una máscara, la máscara a un contorno, el contorno a un vector descriptor y el vector a una decisión de especie.

Los resultados disponibles muestran que, incluso con una base de datos limitada, el proceso alcanza una precisión global del 70 %, con comportamiento diferencial entre especies y una ventaja clara de la segmentación basada en SAM en escenarios complejos. Esto confirma la viabilidad del enfoque y su potencial para fortalecerse con conjuntos de datos más amplios y condiciones de validación más diversas.

Desde la lógica editorial del *Seminario GeoLimna–GEPAR*, este trabajo representa una transición valiosa: pasar de una necesidad operativa de monitoreo en pesca y conservación a un proceso técnico formalizado, documentado, protegido y con proyección internacional.

Agradecimientos

Se agradece a la Universidad de Antioquia, a la Universidad Veracruzana y a los grupos de investigación GeoLimna y GEPAR por el entorno académico e institucional que hizo posible este trabajo. Se reconoce especialmente la participación de Mariana Paniagua Lopera en el desarrollo del proceso, así como el acompañamiento de Juan Pablo Serna López, Karla Cirila Garcés García y David Stephen Fernández Mc Cann. Se agradece además a los estudiantes, docentes e investigadores que contribuyeron con imágenes de campo y con la validación conceptual del procedimiento, así como al espacio brindado por el *World Fisheries Congress* para socializar esta línea de trabajo.

Referencias

- [1] D. S. Fernández Mc Cann, J. P. Serna López y M. Paniagua Lopera, *Solicitud para asignación de puntos por desarrollo de software SHARKSID*, Documento institucional, Universidad de Antioquia, Medellín, Colombia, 2024.
- [2] *Software para la identificación de tiburones – SHARKSID: Manual de usuario*, Universidad de Antioquia, Medellín, Colombia, 2024.
- [3] *Software para la identificación de tiburones – SHARKSID: Manual técnico*, Universidad de Antioquia, Medellín, Colombia, 2024.

- [4] D. S. Fernández Mc Cann y J. P. Serna López, *Certificación de uso del software SHARKSID*, Certificación institucional, Universidad de Antioquia, Medellín, Colombia, jun. de 2024.
- [5] Dirección Nacional de Derecho de Autor, *SHARKSID: Software para identificación de tiburones. Certificado de registro de soporte lógico–software*, Libro 13-98-428, fecha de registro: 29-may-2024, 2024.
- [6] M. Paniagua Lopera, *Identificación de tiburones con métodos de visión artificial*, Trabajo de grado, Ingeniería Electrónica, Medellín, Colombia, 2024.
- [7] K. C. Garcés-García, J. P. Serna López, D. S. Fernández Mc Cann, R. I. Rivera Henao y M. Paniagua Lopera, *A novel application based on the Fourier coefficients to identify sharks*, Resumen presentado al 9th World Fisheries Congress, Seattle, Washington, 2024.
- [8] W. F. Congress, *Certificate of presentation: A novel application based on the Fourier coefficients to identify sharks*, 9th World Fisheries Congress, Seattle, Washington, March 3–7, 2024, 2024.
- [9] L. A. Carrillo-Aguilar, E. Guerra-Rosas, J. Álvarez-Borrego, H. A. Echavarría-Heras y S. Hernández-Muñoz, «Identification of Shark Species Based on Their Dry Dorsal Fins through Image Processing,» *Applied Sciences*, vol. 12, n.º 22, pág. 11 646, 2022. DOI: 10.3390/app122211646.

Capítulo 11

Comparación morfométrica multitemporal de un humedal costero mediante análisis batimétrico y modelación espacial

Autores

Elkin Libardo Ríos¹; Fabio de Jesús Vélez Macías²; David Stephen Fernández Mc Cann³; Valeria Naranjo Morales⁴; Néstor Jaime Aguirre Ramírez⁵.

Resumen

Los humedales costeros de baja profundidad constituyen sistemas altamente sensibles a procesos de colmatación, cambios en la dinámica hídrica y transformaciones del entorno geomorfológico. En este capítulo se presenta una comparación morfométrica multitemporal de la ciénaga El Eneal, ubicada en el municipio de San Onofre, Sucre, Colombia, a partir de información batimétrica levantada en 2004 y 2022. El propósito fue cuantificar la variación de rasgos morfométricos fundamentales del humedal y discutir su significado

¹Magíster en Ingeniería Ambiental, integrante del grupo GeoLimna.

²Profesor titular, Departamento de Ingeniería Ambiental y Sanitaria.

³Profesor, Departamento de Ingeniería Electrónica y Telecomunicaciones.

⁴Ingeniera ambiental, integrante del grupo GeoLimna.

⁵Profesor titular, coordinador del grupo GeoLimna.

^{01,2,3,4,5}Universidad de Antioquia. Grupos de investigación GeoLimna y GEPAR.

en relación con la dinámica del sistema. Para ello se emplearon puntos georreferenciados de profundidad, modelación espacial del terreno mediante una red triangular irregular y estimación comparativa de área, volumen y profundidad máxima. Los resultados muestran que, aunque la profundidad máxima pasó de 2.01 m a 2.8 m, el humedal redujo su superficie total y su volumen almacenado, lo que sugiere una transformación progresiva de su anatomía y una pérdida neta de capacidad de almacenamiento. La comparación entre ambos momentos permite interpretar el sistema no solamente como un cuerpo de agua aislado, sino como un reservorio sometido a un proceso morfodinámico de reducción espacial con implicaciones ecológicas, limnológicas y sociales. El análisis confirma la necesidad de monitoreo continuo y de estrategias de manejo que reconozcan el valor del humedal como reserva de agua dulce y como componente estructural del paisaje costero del golfo de Morrosquillo.

Palabras clave: humedal costero; morfometría; batimetría; colmatación; modelación espacial; ciénaga El Eneal.

11.1. Introducción

Los rasgos morfométricos de un cuerpo de agua no constituyen una simple descripción geométrica de su forma. Expresan, en términos físicos, procesos ecológicos, sedimentarios e hidrológicos que se manifiestan en el espacio y en el tiempo. Cuando un humedal pierde área, cuando su vaso cambia de forma o cuando su volumen se reduce, esos cambios alteran la capacidad de almacenamiento, modifican la interacción entre sedimento y columna de agua, afectan la circulación interna y transforman el papel ecológico del sistema dentro del paisaje que lo contiene.

La ciénaga El Eneal, ubicada en el municipio de San Onofre, en el golfo de Morrosquillo, constituye un caso particularmente relevante. El artículo base de este capítulo la describe como un humedal costero de importancia ecológica y ambiental, además de una de las principales reservas de agua dulce potencialmente utilizable en la zona. El estudio comparó información batimétrica obtenida en 2004 con nuevos datos levantados en 2022, con el fin de evaluar la variación de los parámetros morfométricos más relevantes del sistema. Los resultados reportados muestran que en 2004 la ciénaga tenía un área de 474,311,34 m², un volumen de 550,939,50 m³ y una profundidad máxima de 2,01 m, mientras que en 2022 presentó un área de 353,760,2 m², un volumen de 401,024,3 m³ y una profundidad máxima de 2,8 m. Esta comparación llevó a concluir que el sistema perdió área y volumen de manera significativa.

Este comportamiento no se deja resumir únicamente por una disminución superficial. El

hecho de que la profundidad máxima aumente al tiempo que el área y el volumen disminuyen obliga a una lectura más cuidadosa. No se trata de un simple secado homogéneo, sino de una reconfiguración de la geometría interna del humedal. Esa reconfiguración puede asociarse con colmatación en sectores litorales, concentración del agua en zonas residuales más profundas y reducción progresiva del espejo de agua efectivo.

Este capítulo retoma ese trabajo y lo reformula dentro de la lógica editorial del *Seminario GeoLimna-GEPAR*. El énfasis no está solamente en el resultado numérico final, sino en el proceso de comparación morfométrica como herramienta para interpretar el comportamiento físico de un humedal estratégico.

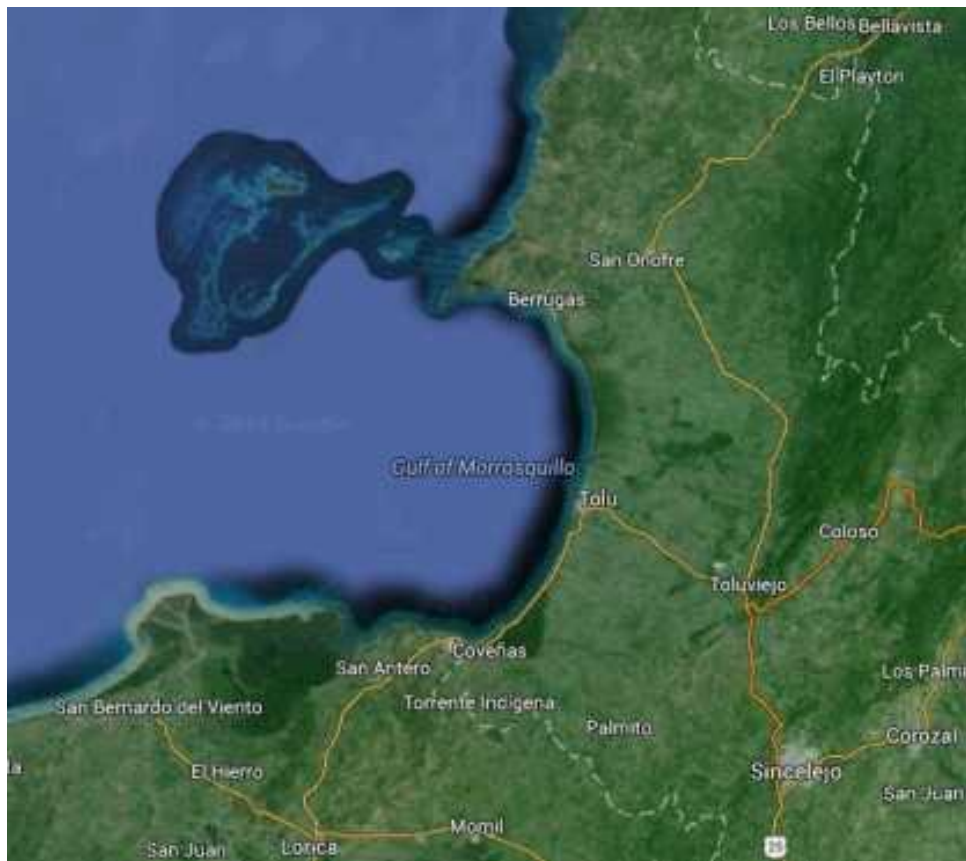


Figura 11.1: Localización regional de la ciénaga El Eneal y su relación espacial con San Onofre y la franja costera del golfo de Morrosquillo.

11.2. Marco físico del problema

La comprensión de un humedal requiere reconocer, en primer lugar, que su geometría es inseparable de su contexto ambiental. El Eneal forma parte del sistema de humedales de

la reserva Sanguaré y se encuentra en una llanura costera con marcada sensibilidad al aporte de sedimentos, a la escorrentía superficial y a las variaciones climáticas regionales. El artículo señala que el cuerpo de agua se ubica en el golfo de Morrosquillo, una región asociada a una dinámica deltaica, y recuerda además que la ciénaga ha evolucionado desde una condición estuarina hacia un sistema de agua dulce.

Esa historia geomorfológica importa porque condiciona la anatomía actual del vaso. Un humedal que transitó desde una influencia marina hacia un estado más limnético conserva, por lo general, huellas de esa transición en su forma, en su interacción con el sedimento y en la distribución espacial de profundidades. De hecho, el mismo artículo asocia la relevancia ambiental del sistema con su función como reservorio de agua dulce en una zona donde este recurso es escaso.

Desde el punto de vista físico, puede representarse el fondo del humedal mediante una función batimétrica

$$z = b(x, y),$$

donde x y y son las coordenadas horizontales en un sistema de proyección plano, y $b(x, y)$ es la cota del fondo o, de manera equivalente, la profundidad medida respecto a un plano de referencia. Esta formulación es la base para transformar un conjunto discreto de puntos medidos en una superficie continua a partir de la cual pueden estimarse magnitudes integrales como área y volumen.

Si $\Omega \subset \mathbb{R}^2$ representa el dominio espacial ocupado por el espejo de agua, entonces el área superficial del humedal se expresa como

$$A = \iint_{\Omega} 1 \, d\Omega.$$

Esta ecuación indica que el área es la medida del dominio ocupado por la lámina de agua. Una vez se dispone de una superficie batimétrica, la estimación del volumen contenido puede expresarse como

$$V = \iint_{\Omega} h(x, y) \, d\Omega,$$

donde $h(x, y)$ representa la profundidad local del agua respecto al plano de referencia adoptado para el cálculo. En términos prácticos, esta formulación es la que subyace al uso de herramientas de modelación espacial como el método *Surface Volume* mencionado en el artículo.

11.3. Datos, levantamiento y representación espacial

La comparación realizada en el estudio se apoyó en dos campañas de observación separadas por dieciocho años. Para 2004 se consideraron 93 puntos de profundidad distribuidos sobre el espejo de agua, mientras que para 2022 se levantaron 95 puntos georreferenciados. El artículo incluye las tablas completas con las coordenadas y profundidades para ambos años, así como las figuras con la ubicación espacial de los puntos seleccionados para el análisis batimétrico.

La diferencia entre ambos conjuntos no invalida la comparación. Muestra, más bien, que el análisis trabaja con muestreos discretos de la geometría del vaso, no con una observación continua del fondo. Esto obliga a un paso intermedio de reconstrucción espacial. En el trabajo base se reporta explícitamente el uso de un modelo TIN, es decir, una red triangular irregular, dentro de un sistema de proyección UTM. Un TIN puede entenderse como una partición del dominio Ω en triángulos no superpuestos

$$\Omega = \bigcup_{i=1}^{N_T} T_i,$$

donde cada triángulo T_i tiene como vértices puntos batimétricos medidos en campo. Sobre cada triángulo se asume una interpolación lineal de la profundidad, lo que permite construir una aproximación continua de $b(x, y)$.

La utilidad de esta representación es decisiva. Si los puntos medidos quedaran solamente como una nube discreta, la comparación entre años se limitaría a contrastar profundidades puntuales. Al interpolar mediante TIN, es posible estimar no solo la profundidad local sino también la forma global del vaso, el contorno del espejo de agua y el volumen contenido bajo una cota de referencia.



Figura 11.2: Distribución espacial de los puntos batimétricos utilizados para la reconstrucción del vaso en 2004 y 2022.

11.4. Metodología comparativa

El procedimiento seguido en este trabajo puede describirse como una secuencia de cinco etapas. Primero, se levantaron o consolidaron los puntos georreferenciados de profundidad para cada año de análisis. Segundo, se proyectaron en un mismo sistema de coordenadas. Tercero, se construyó una superficie batimétrica interpolada mediante TIN. Cuarto, se definió una cota de referencia común para ambos momentos, reportada en el artículo como -1 , sobre la cual se estimaron el área y el volumen. Finalmente, se compararon los parámetros resultantes y se interpretaron sus diferencias.

Si denotamos por

$$\mathcal{P}_{2004} = \{(x_i, y_i, h_i)\}_{i=1}^{93}$$

el conjunto de puntos batimétricos del año 2004 y por

$$\mathcal{P}_{2022} = \{(x_j, y_j, h_j)\}_{j=1}^{95}$$

el conjunto del año 2022, entonces el paso de interpolación puede entenderse como la

construcción de dos funciones aproximadas:

$$h_{2004}(x, y), \quad h_{2022}(x, y).$$

A partir de ellas se calculan

$$A_{2004}, V_{2004}, D_{\text{máx},2004}$$

y

$$A_{2022}, V_{2022}, D_{\text{máx},2022}.$$

La variación entre ambos estados se resume mediante diferencias simples:

$$\Delta A = A_{2022} - A_{2004},$$

$$\Delta V = V_{2022} - V_{2004},$$

$$\Delta D_{\text{máx}} = D_{\text{máx},2022} - D_{\text{máx},2004}.$$

Estas diferencias tienen un valor interpretativo inmediato. Un $\Delta A < 0$ indica reducción del espejo de agua. Un $\Delta V < 0$ indica pérdida de capacidad de almacenamiento. Un $\Delta D_{\text{máx}} > 0$, en cambio, no implica mejora del sistema, sino una redistribución interna de la profundidad que debe ser leída junto con las otras dos magnitudes.

11.5. Resultados morfométricos

Los resultados principales del artículo pueden resumirse con claridad. Para el año 2004, la ciénaga El Eneal presentó:

$$A_{2004} = 474,311,34 \text{ m}^2,$$

$$V_{2004} = 550,939,50 \text{ m}^3,$$

$$D_{\text{máx},2004} = 2,01 \text{ m}.$$

Para el año 2022, los valores reportados fueron:

$$A_{2022} = 353,760,2 \text{ m}^2,$$

$$V_{2022} = 401,024,3 \text{ m}^3,$$

$$D_{\text{máx},2022} = 2,8 \text{ m}.$$

Al aplicar las expresiones diferenciales, se obtiene:

$$\Delta A = 353,760,2 - 474,311,34 = -120,551,14 \text{ m}^2,$$

$$\Delta V = 401,024,3 - 550,939,50 = -149,915,2 \text{ m}^3,$$

$$\Delta D_{\text{máx}} = 2,8 - 2,01 = 0,79 \text{ m}.$$

Este conjunto de resultados muestra una señal clara: la ciénaga perdió superficie y volumen, aunque al mismo tiempo incrementó su profundidad máxima. Dicho en términos geométricos, el humedal no se hizo simplemente menos profundo; se hizo más pequeño y más concentrado, con sectores probablemente más deprimidos que retienen agua mientras otras zonas se somerizan, colmatan o quedan fuera del espejo activo.

El artículo insiste justamente en esta lectura al afirmar que el reservorio se está colmatando y que, de mantenerse la tendencia, podría desaparecer en los próximos años. Esa interpretación se refuerza al observar las figuras temporales del artículo, en las cuales se aprecia una transformación visible de la forma y extensión del cuerpo de agua entre 2006, 2010, 2020 y 2023.

Tabla 11.1: Comparación de parámetros morfométricos de la ciénaga El Eneal entre 2004 y 2022

Parámetro	2004	2022	Variación
Área (m ²)	474311.34	353760.2	-120551.14
Volumen (m ³)	550939.50	401024.3	-149915.2
Profundidad máxima (m)	2.01	2.8	+0.79

11.6. Interpretación geomorfológica y limnológica

La pérdida simultánea de área y volumen, acompañada de un aumento de la profundidad máxima, exige una interpretación más cuidadosa que la de un simple proceso lineal de reducción. Si el sistema hubiera experimentado únicamente una colmatación homogénea del fondo, sería razonable esperar una disminución de la profundidad máxima junto con la reducción de volumen. Pero aquí ocurre otra cosa: el humedal parece haber perdido sectores litorales o superficies someras mientras retiene, en zonas puntuales, una profundidad mayor.

Esto sugiere una reorganización espacial del vaso. En términos conceptuales, puede decirse

que la función batimétrica

$$h(x, y)$$

no cambió solo en magnitud media, sino en distribución. Algunas zonas del dominio Ω probablemente desaparecieron del espejo de agua, mientras otras conservaron o acentuaron su papel como depresiones principales del sistema. Ese comportamiento es coherente con procesos de sedimentación diferencial, invasión o consolidación del borde por vegetación y concentración del agua en zonas residuales más profundas.

El artículo describe además que El Eneal es un cuerpo de agua poco profundo, con mezcla probablemente favorecida por la acción del viento, y señala que su línea de costa no es tortuosa sino relativamente bien definida por vegetación de *Typha* o enea. Ese borde vegetal, que en apariencia estabiliza la forma, también puede ser leído como una frontera viva que participa de la transformación del sistema, en la medida en que consolida sectores litorales y acompaña el avance de la colmatación.

La comparación temporal también adquiere peso limnológico. Un sistema más pequeño y con menor volumen responde de manera distinta a la evaporación, al aporte de nutrientes, al pulso de lluvias y a las entradas salinas estacionales. En otras palabras, la pérdida morfométrica no es solo una variación de forma; es una alteración del régimen funcional del humedal.

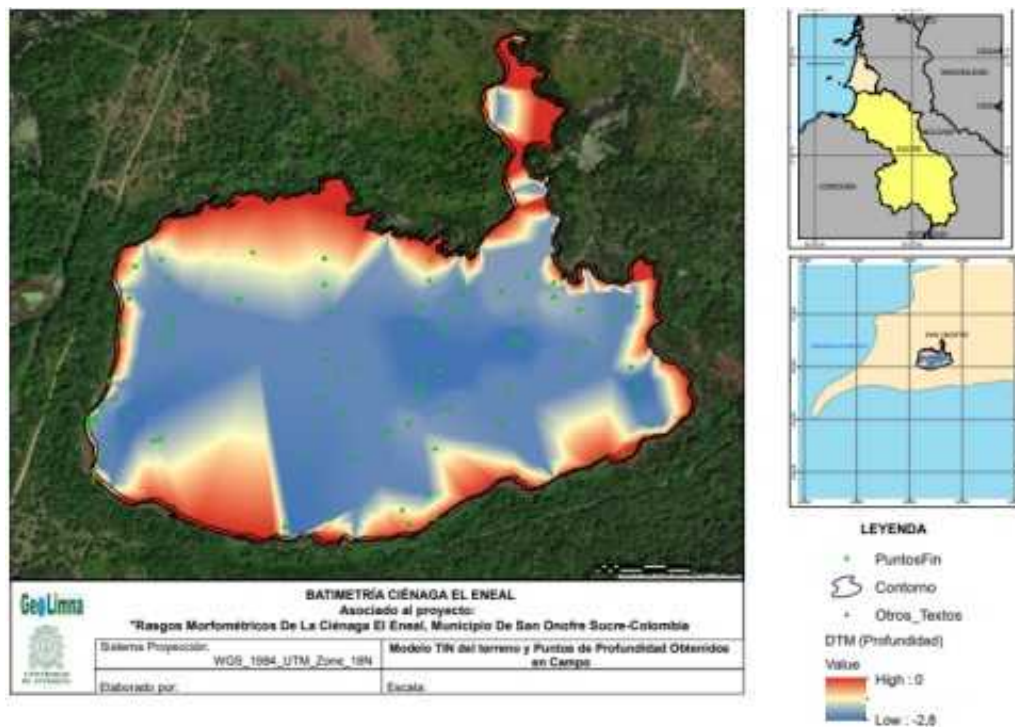


Figura 11.3: Mapas batimétricos comparativos del humedal en los dos periodos de análisis, mostrando la redistribución espacial de la profundidad.

11.7. Evidencia visual del cambio

Una de las fortalezas del artículo base es que no se limita a tablas y valores agregados. También aporta evidencia visual. En una página intermedia se presenta una vista panorámica actual del humedal, mientras que en otra figura se reúnen imágenes de 2006, 2010, 2020 y 2023 tomadas desde plataformas de observación remota. Esa secuencia temporal permite ver que la reducción del cuerpo de agua no es una inferencia puramente matemática, sino un fenómeno visible en la evolución de la cobertura superficial.

La lectura conjunta de esas imágenes y de los resultados morfométricos fortalece el argumento del capítulo. Las ecuaciones resumen la variación; las imágenes la hacen tangible. Desde una perspectiva metodológica, esto es importante porque los humedales suelen ser discutidos solo desde inventarios puntuales o descripciones cualitativas. Aquí, en cambio, la anatomía del sistema se reconstruye con números y se confirma con imágenes.

Si se quisiera formalizar este tipo de contraste multitemporal de manera general, podría

definirse una familia de dominios observados

$$\Omega(t_1), \Omega(t_2), \dots, \Omega(t_n),$$

y asociar a cada uno una serie de métricas morfométricas

$$\mathcal{M}(t) = \{A(t), V(t), D_{\text{máx}}(t)\}.$$

La comparación entre tiempos deja entonces de ser anecdótica y se convierte en una trayectoria geométrica del sistema.

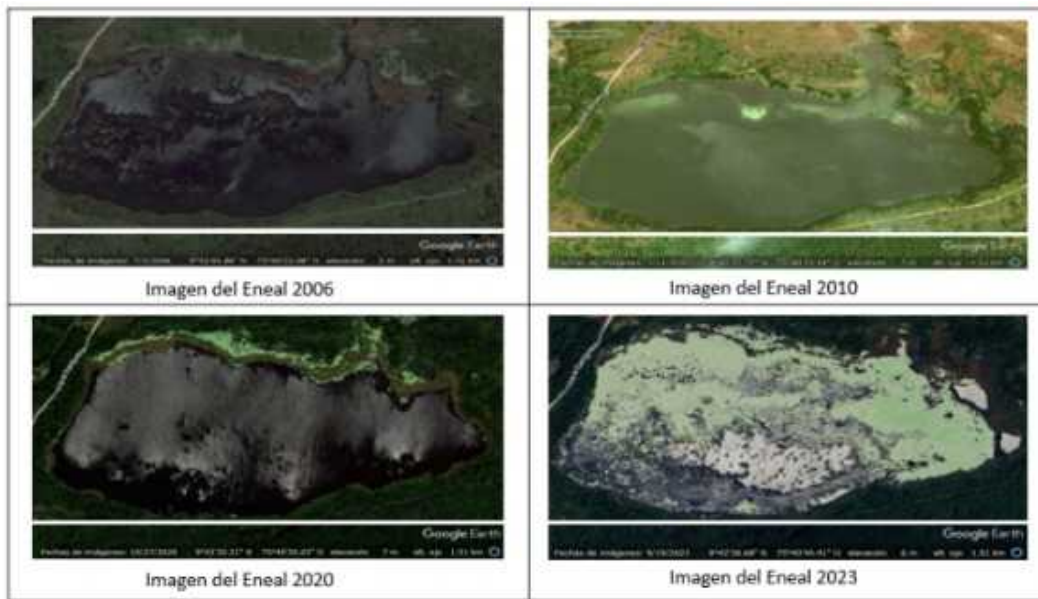


Figura 11.4: Secuencia multitemporal del humedal, mostrando la transformación progresiva del espejo de agua en diferentes años.

11.8. Discusión

Este estudio tiene un valor especial porque muestra cómo una comparación morfométrica relativamente simple puede revelar procesos de transformación ecológicamente serios. No se recurrió a modelos complejos ni a sensores sofisticados fuera del alcance universitario. Se trabajó con puntos georreferenciados, reconstrucción espacial del fondo y comparación de parámetros bien definidos. Y, sin embargo, el resultado es contundente: el humedal ha perdido una porción importante de su superficie y de su volumen.

En el marco del seminario, este capítulo también tiene una importancia metodológica adicional. Muestra una forma de articular geografía, limnología, levantamiento de campo

y modelación espacial sin diluir el fenómeno en exceso teórico. La matemática aquí no aparece como ornamento, sino como una manera de expresar lo que físicamente ocurrió con el sistema.

También hay una lectura más amplia. Cuando un humedal costero como El Eneal se reduce, no pierde solo agua. Pierde resiliencia frente a sequías, cambia su relación con el paisaje, altera el hábitat disponible para organismos acuáticos y disminuye su capacidad como reserva hídrica. En regiones con escasez de agua dulce, esa transformación tiene un peso social directo.

El artículo original concluye precisamente con una advertencia fuerte: si el proceso continúa, la ciénaga podría desaparecer en los próximos años. Esa afirmación, lejos de ser retórica, está sostenida por una trayectoria morfométrica clara y por evidencia visual temporal.

11.9. Conclusiones

La comparación morfométrica multitemporal realizada sobre la ciénaga El Eneal mostró que entre 2004 y 2022 el humedal experimentó una reducción significativa de su área superficial y de su volumen almacenado, con una variación simultánea de la profundidad máxima. Los resultados indican una pérdida de 120,551,14 m² de superficie y de 149,915,2 m³ de volumen, lo que evidencia una transformación sustancial de la anatomía del vaso.

La formulación matemática permitió expresar con claridad la transición entre el levantamiento discreto de profundidades y la obtención de magnitudes integrales como área y volumen. A su vez, la comparación entre superficies interpoladas puso de manifiesto que el cambio no corresponde a una simple reducción uniforme, sino a una reorganización espacial del humedal compatible con procesos de colmatación y concentración de la lámina de agua en sectores más profundos.

Desde el punto de vista ambiental, el hallazgo es especialmente relevante porque El Eneal constituye una reserva estratégica de agua dulce en la punta norte del golfo de Morrosquillo. Su transformación no es un asunto únicamente geométrico, sino un problema ecológico y social. En consecuencia, este capítulo respalda la necesidad de establecer un seguimiento morfométrico periódico del humedal e incorporar esta clase de análisis dentro de estrategias más amplias de gestión, conservación y restauración.

Agradecimientos

Se agradece a la Universidad de Antioquia y al grupo de investigación GeoLimna por el soporte académico y científico que hizo posible este trabajo. Se reconoce también el valor del proyecto sobre la evolución morfométrica e hidrobiológica de la ciénaga El Eneal como marco integrador de la investigación. Finalmente, se agradece a los administradores de la Reserva Sanguaré y a quienes facilitaron el trabajo de campo y la continuidad del monitoreo del sistema.

Referencias

- [1] E. L. Ríos, F. Vélez Macías, D. S. Fernández Mc Cann, V. Naranjo Morales y N. J. Aguirre Ramírez, «Comparación de rasgos morfométricos de la ciénaga El Eneal, municipio de San Onofre Sucre, Colombia,» *Producción + Limpia*, vol. 20, n.º 1, 2025. DOI: 10.22507/pm1.v20n1a3629.
- [2] E. Ríos, J. Palacio y N. Aguirre, «Variabilidad fisicoquímica del agua en la ciénaga El Eneal, reserva natural Sanguaré, municipio de San Onofre, Sucre, Colombia,» *Revista Facultad de Ingeniería*, n.º 46, págs. 39-45, 2008.
- [3] E. Ríos, J. Palacio y N. Aguirre, «Productividad primaria y sustancias húmicas en la ciénaga El Eneal, San Onofre, Sucre, Colombia,» *Revista Facultad de Ingeniería*, n.º 47, págs. 67-72, 2009.
- [4] E. L. Ríos Ortiz, F. d. J. Vélez Macías, D. S. Fernández Mc Cann, V. Naranjo Morales, M. G. Vallejo y N. J. Aguirre Ramírez, «Primer registro de *Utricularia* sp. en el humedal El Eneal, San Onofre, Colombia,» *Producción + Limpia*, vol. 18, n.º 2, 2023. DOI: 10.22507/pm1.v18n2a4.
- [5] Y. Montoya, F. Vélez y N. Aguirre, «Características morfométricas de un lago de plano inundable tropical (ciénaga Hoyo Los Bagres, Colombia),» *Revista Facultad de Ingeniería Universidad de Antioquia*, n.º 59, págs. 203-214, 2011.
- [6] W. J. Mitsch y J. G. Gosselink, *Wetlands*. John Wiley & Sons, 2000.
- [7] UNESCO, *Convención relativa a los humedales de importancia internacional especialmente como hábitat de especies acuáticas*, Convención Ramsar, 1971.
- [8] G. Roldán, «Revisión histórica de la limnología en Colombia,» *Revista de la Academia Colombiana de Ciencias Exactas, Físicas y Naturales*, vol. 44, n.º 171, págs. 303-328, 2020. DOI: 10.18257/raccefyn.1056.

- [9] M. Camino, M. Bó, J. Cionchi, A. López de Armentia, J. Del Río y S. De Marco, «Estudio morfométrico de las cuencas de drenaje de la vertiente sur del sudeste de la provincia de Buenos Aires (Argentina),» *Revista Universitaria de Geografía*, vol. 27, n.º 1, págs. 73-97, 2018.
- [10] F. Patiño y F. Amaya, *Estudio ecológico del Golfo de Morrosquillo*. Universidad Nacional de Colombia, 1993.

Capítulo 12

Caracterización ecohidrobiológica de un humedal costero y registro de una planta carnívora acuática en El Eneal, San Onofre

Autores

Elkin Libardo Ríos Ortiz¹; Fabio de Jesús Vélez Macías²; David Stephen Fernández Mc Cann³; Valeria Naranjo Morales⁴; Melanie Gabriela Vallejo⁵; Néstor Jaime Aguirre Ramírez⁶.

Resumen

Los humedales costeros tropicales de agua dulce constituyen sistemas altamente sensibles a variaciones hidrológicas, a la disponibilidad de nutrientes y a cambios en la estructura biológica de sus comunidades. En este capítulo se presenta una caracterización ecohidrobiológica del humedal El Eneal, ubicado en el golfo de Morrosquillo, Colombia, con énfasis

¹Profesor e investigador, integrante del grupo GeoLimna.

²Profesor titular, Departamento de Ingeniería Ambiental y Sanitaria.

³Profesor, Departamento de Ingeniería Electrónica y Telecomunicaciones, integrante de GEPAR y GeoLimna.

⁴Estudiante de Ingeniería Sanitaria y Ambiental.

⁵Estudiante de Ingeniería Sanitaria y Ambiental.

⁶Profesor titular, coordinador del grupo GeoLimna.

^{01.2.3.4.5.6}Universidad de Antioquia. Grupos de investigación GeoLimna y GEPAR.

en el hallazgo de una planta carnívora acuática del género *Utricularia*, no reportada previamente para este sistema. La investigación integra información histórica de nutrientes y productividad primaria con mediciones realizadas en 2022, además de registros de flora acuática, fitoplancton, zooplancton y macroinvertebrados asociados. El trabajo muestra que el humedal mantiene concentraciones bajas de nitrógeno y fósforo a lo largo del tiempo, y que la productividad primaria bruta medida en 2022 fue inferior a la reportada en 2004. En ese contexto, la presencia simultánea de *Najas guadalupensis* y *Utricularia sp.* sugiere una relación ecológica relevante entre la estructura vegetal sumergida, la captura de organismos planctónicos y la baja productividad del sistema. Más allá del valor taxonómico del registro, el hallazgo aporta elementos para comprender mejor la dinámica ecológica de un humedal estratégico para la oferta hídrica en la región norte del golfo de Morrosquillo.

Palabras clave: humedal; productividad primaria; planta acuática; planta carnívora; *Utricularia*; ecohidrobiología.

12.1. Introducción

Los humedales tropicales de baja profundidad son sistemas donde los procesos físicos, químicos y biológicos no pueden analizarse por separado. Su poca profundidad favorece una interacción intensa entre sedimento, columna de agua, vegetación acuática y organismos planctónicos. Esa condición los vuelve especialmente sensibles a cambios ambientales, pero también los convierte en escenarios adecuados para observar relaciones ecológicas finas que en otros cuerpos de agua pasarían inadvertidas.

El humedal El Eneal representa un caso de especial interés en la región del golfo de Morrosquillo. El artículo base lo describe como una reserva superficial de agua dulce muy valiosa para una zona donde el acceso a agua utilizable es limitado, y lo sitúa como parte del proyecto más amplio sobre la evolución morfométrica e hidrobiológica del sistema. En el marco de ese proyecto, durante los muestreos de julio de 2022 se encontró por primera vez la presencia de una planta carnívora del género *Utricularia*, hecho que no había sido reportado previamente ni para El Eneal ni para los humedales de esta parte del golfo. [1]

Ese hallazgo no es únicamente una novedad florística. Tiene implicaciones ecológicas directas. El mismo artículo plantea que la presencia simultánea de *Najas guadalupensis* y *Utricularia sp.*, junto con los bajos valores de productividad primaria observados en el humedal, permite plantear una hipótesis funcional: la baja producción no sólo estaría asociada a la condición oligotrófica del sistema, sino también a la presión biológica ejercida por macrófitas sumergidas y por una planta capaz de capturar organismos del plancton.

[1]

Este capítulo reorganiza ese artículo desde la lógica del seminario. El énfasis no está solo en el primer registro, sino en el proceso de caracterización ecohidrobiológica que permitió relacionar calidad del agua, productividad primaria, estructura vegetal y biodiversidad microscópica con la presencia de la planta carnívora.



Figura 12.1: Vista aérea del área de estudio y su contexto costero en el norte del golfo de Morrosquillo.

12.2. Planteamiento ecológico del problema

La pregunta que subyace a este trabajo puede formularse así: ¿cómo interpretar la presencia de una planta carnívora acuática en un humedal costero de agua dulce que, además, presenta bajas concentraciones de nutrientes y baja productividad primaria?

Responder a esa pregunta exige articular varias dimensiones del sistema. Sea

$$\mathcal{H} = (\mathcal{Q}, \mathcal{P}, \mathcal{B}),$$

donde \mathcal{Q} representa el estado químico del agua, \mathcal{P} la productividad primaria y \mathcal{B} la estructura biológica observada en el humedal. La hipótesis ecológica implícita del artículo puede expresarse como una dependencia funcional:

$$\mathcal{P} = f(\mathcal{Q}, \mathcal{M}, \mathcal{T}),$$

donde \mathcal{M} representa la comunidad de macrófitas acuáticas y \mathcal{T} la disponibilidad trófica

de organismos planctónicos potencialmente capturables.

Esta ecuación no pretende ser un modelo mecanicista completo. Su valor aquí es organizador. Indica que la productividad primaria no depende únicamente de los nutrientes disueltos en el agua, sino también de interacciones bióticas que modifican la disponibilidad de recursos y la estructura del sistema.

En ese sentido, la presencia de *Utricularia sp.* no es un dato accesorio. Es un elemento que reconfigura la lectura del humedal, porque introduce un mecanismo de captura biológica de presas pequeñas en un ambiente de baja disponibilidad de nutrientes.

12.3. Área de estudio y contexto ambiental

El artículo ubica el humedal El Eneal en la finca La Aguada, municipio de San Onofre, Sucre, en la zona de Sanguaré, dentro del golfo de Morrosquillo. También recuerda que la precipitación media anual es del orden de 1400 mm, que las lluvias se concentran en mayo, junio, agosto, septiembre y octubre, y que el viento medio anual es de 3.3 m/s. [1]

Estas condiciones son importantes porque ayudan a comprender el carácter fluctuante del sistema. La alternancia entre épocas secas y lluviosas influye en el nivel del agua, la concentración de nutrientes, la transparencia, la extensión de la zona fótica y la estructura de las comunidades acuáticas. En humedales poco profundos como El Eneal, esos factores no sólo cambian la física del sistema, sino que alteran también el espacio ecológico disponible para macrófitas, plancton y macroinvertebrados.

Si se representa la disponibilidad de luz en la columna de agua mediante una función decreciente con la profundidad,

$$I(z) = I_0 e^{-kz},$$

donde I_0 es la irradiancia superficial, z la profundidad y k el coeficiente de atenuación, entonces el límite de la zona fótica depende directamente de la capacidad del sistema para transmitir luz. El trabajo reporta precisamente mediciones del límite de la zona fótica y de la profundidad en 2022, lo que permite conectar productividad primaria con disponibilidad lumínica. [1]



Figura 12.2: Vista general del humedal El Eneal y del espejo de agua correspondiente al sistema de estudio.

12.4. Diseño metodológico y estructura del muestreo

La metodología empleada integró tres bloques: calidad del agua, productividad primaria y biodiversidad. Esta organización es conceptualmente acertada porque evita reducir el análisis a una sola dimensión. Un humedal puede tener bajos nutrientes, pero ello sólo adquiere sentido ecológico cuando se cruza con la respuesta biológica del sistema.

En el componente químico, se tomó una muestra integrada de agua en el centro de la ciénaga usando una botella tipo Kemmerer. Se midieron nitratos, amonio, ortofosfatos y dureza. En el componente hidrobiológico, se realizaron arrastres circulares sobre el espejo de agua durante diez minutos con redes de $10 \mu\text{m}$ para fitoplancton y $40 \mu\text{m}$ para zooplancton. Para el componente de productividad primaria se utilizó el método de botellas clara y oscura, junto con mediciones de oxígeno disuelto usando una sonda multiparamétrica. Finalmente, para el componente de plantas acuáticas se dispusieron transectas de 10 m desde la orilla hacia el espejo de agua, con cuadrantes de 1 m^2 a distintas distancias para estimar biomasa húmeda y macroinvertebrados asociados. [1]

Esta lógica de muestreo puede resumirse como una secuencia:

$$\mathcal{E} = \mathcal{N} \circ \mathcal{O} \circ \mathcal{V},$$

donde \mathcal{V} representa el levantamiento de variables fisicoquímicas, \mathcal{O} la observación hidrobiológica y \mathcal{N} el registro de la vegetación acuática y de los organismos asociados.

El valor de esta estructura está en que no aísla el hallazgo de *Utricularia sp.* como un evento puramente taxonómico, sino que lo sitúa dentro de una matriz de observaciones ambientales concurrentes.

12.5. Nutrientes y condición trófica del sistema

Uno de los argumentos centrales del artículo es la persistencia de bajas concentraciones de nutrientes en El Eneal a lo largo del tiempo. Las tablas históricas de 2003 y 2004 muestran valores pequeños de nitrato, nitrito, amonio y ortofosfatos, y los datos de 2022 indican que, casi dos décadas después, el nitrógeno y el fósforo continúan en rangos bajos. [1]

Si denotamos por

$$N = [NO_3^-, NO_2^-, NH_4^+]$$

el vector de formas principales del nitrógeno inorgánico disuelto, y por

$$P = [PO_4^{3-}]$$

el contenido de fósforo soluble, entonces el estudio sugiere que la magnitud conjunta

$$\|[N, P]\|$$

se mantiene reducida en comparación con sistemas más productivos. Desde una perspectiva limnológica clásica, esta condición es consistente con un humedal oligotrófico.

El artículo invoca precisamente la clasificación de Vollenweider para interpretar el sistema como oligotrófico. Esta conclusión es importante porque ofrece el trasfondo químico sobre el cual debe entenderse el hallazgo de la planta carnívora: en ambientes con nutrientes bajos, la captura biológica de presas puede convertirse en una estrategia relevante para suplir necesidades nutricionales.

Tabla 12.1: Síntesis comparativa de nutrientes medidos en El Eneal

Periodo	Nitrato	Nitrito	Amonio	Fosfato/Ortofosfato
2003	Bajo	Bajo	Bajo	Bajo
2004	Bajo	Bajo	Bajo	Bajo
2022	Bajo	Bajo / no destacado	Bajo	Bajo

12.6. Productividad primaria y disponibilidad de luz

La productividad primaria fue estimada mediante el método de botellas clara y oscura, una estrategia clásica que permite inferir producción a partir de variaciones de oxígeno disuelto. Si $O_2(i)$ representa la concentración inicial de oxígeno y $O_2(f)$ la final, la diferencia entre ambas, controlada por el tiempo de incubación y por el comportamiento de las botellas clara y oscura, permite estimar la producción neta y la respiración.

En forma general, puede expresarse una tasa de producción como

$$PPB = \frac{\Delta O_2}{\Delta t} \cdot \alpha,$$

donde ΔO_2 representa la variación corregida de oxígeno, Δt el tiempo de incubación y α un factor de conversión hacia carbono fijado.

El artículo recuerda que en 2004 se habían reportado valores de 42,25 y 48,75 mgC/m³/h, mientras que en julio de 2022 la productividad primaria bruta encontrada fue de

$$PPB_{2022} = 25,12 \text{ mgC/m}^3/\text{h}.$$

Esa disminución refuerza la lectura de un sistema oligoproductivo. [1]

Además, se midió el límite de la zona fótica tanto en el centro como en el litoral, con valores reportados de 2.16 y 2.37, respectivamente, junto con profundidades de 1.80 m y 1.00 m para esos puntos. Esta relación sugiere que una parte importante de la columna de agua permanece dentro de la franja iluminada, de modo que la baja productividad no parece explicarse simplemente por una limitación severa de luz, sino por una combinación más compleja de factores químicos y biológicos. [1]

12.7. Registro de *Utricularia sp.* y significado ecológico

El hallazgo más relevante del trabajo fue la presencia, en julio de 2022, de una planta carnívora del género *Utricularia*. El artículo la presenta como el primer registro de esta planta para el humedal El Eneal y para los humedales de esa parte del golfo de Morrosquillo. [1]

Este dato cobra mayor sentido cuando se lo interpreta dentro de la biología del grupo. Las especies de *Utricularia* son conocidas por poseer estructuras de captura capaces de atrapar pequeños organismos acuáticos, especialmente dentro del espectro zooplanctónico

y microinvertebrado. En ambientes con baja disponibilidad de nutrientes, esta estrategia puede considerarse una vía complementaria de adquisición de nitrógeno y fósforo.

Si se define de manera esquemática la nutrición efectiva de la planta como

$$\mathcal{N}_{eff} = \mathcal{N}_{disuelto} + \mathcal{N}_{captura},$$

entonces la segunda componente, correspondiente a la captura biológica de presas, adquiere relevancia precisamente cuando la primera es limitada. Esta relación explica por qué el hallazgo de *Utricularia* resulta ecológicamente tan sugerente en un sistema oligotrófico.

El artículo conecta esta presencia con la baja productividad primaria del humedal y sugiere que la captura de plancton y fitoplancton por parte de la planta podría contribuir a explicar, al menos parcialmente, los bajos niveles productivos observados. Esta hipótesis no debe tomarse como una demostración causal cerrada, pero sí como una interpretación ecológicamente razonable y fértil para futuras investigaciones. [1]

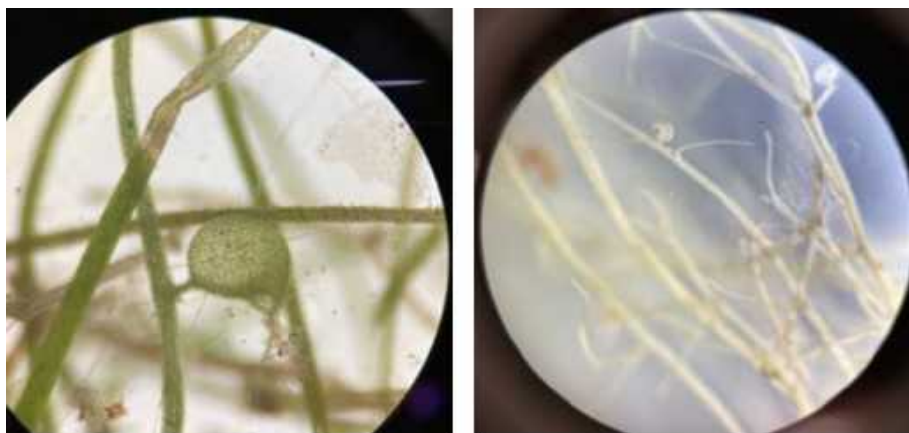


Figura 12.3: Registro microscópico de *Utricularia* sp. hallada en El Eneal.

12.8. Fitoplancton, zooplancton y lectura biológica del sistema

Las observaciones microscópicas reportadas en el artículo muestran una comunidad fitoplanctónica y zooplanctónica muy poco abundante, con registros puntuales de *Euglena* sp., *Trachelomonas* sp., flóculos orgánicos, bacterias filamentosas y restos vegetales en suspensión. [1]

Este patrón es coherente con la interpretación general del humedal. Un sistema con baja

productividad primaria, bajos nutrientes y presencia de macrófitas sumergidas no favorece necesariamente una alta biomasa planctónica libre. Además, la propia presencia de *Utricularia sp.* introduce una forma de depredación sobre microfauna que puede influir en la estructura de la comunidad.

Si la biomasa planctónica disponible se denota por

$$B_p = B_{phyto} + B_{zoo},$$

entonces su variación temporal puede verse afectada no sólo por la entrada de nutrientes sino también por términos de consumo y captura:

$$\frac{dB_p}{dt} = I - C_m - C_u - L,$$

donde I es el ingreso neto por crecimiento y regeneración, C_m el consumo o interferencia asociado a macrófitas, C_u la captura por *Utricularia*, y L otras pérdidas del sistema.

Nuevamente, esta ecuación no pretende modelar cuantitativamente el humedal con precisión mecanicista, pero sí ordenar la lectura ecológica del trabajo: la baja biomasa y la baja productividad pueden emerger de una combinación de escasos nutrientes y fuertes interacciones bióticas.



Figura 12.4: Observaciones microscópicas de fitoplancton, flóculos y restos orgánicos en el sistema.

12.9. Discusión

La principal virtud de este trabajo es que no trata el hallazgo de *Utricularia sp.* como una curiosidad aislada. Lo inserta dentro de una lectura ecohidrobiológica más amplia, en la que convergen la química del agua, la productividad primaria, la estructura de las

macrófitas y la composición observada del plancton. Esa articulación es precisamente la que da peso científico al registro.

Desde la perspectiva del seminario, este capítulo también resulta valioso porque representa una forma de ciencia integradora. No parte de grandes dispositivos instrumentales ni de análisis moleculares complejos, sino de un trabajo de campo bien estructurado, mediciones clásicas bien interpretadas y una mirada ecológica sensible a los detalles del sistema. La matemática entra aquí no como formalismo excesivo, sino como herramienta para ordenar la interpretación de la relación entre recursos, productividad y captura biológica.

Otro punto importante es la ampliación del rango de distribución del género en la región. El artículo muestra, además, una tabla de registros de *Utricularia* en distintos departamentos del país, lo que permite situar a El Eneal dentro de una geografía botánica más amplia. Pero el interés no es meramente corológico. Lo decisivo es que en este humedal específico la presencia de la planta se enlaza con un estado trófico bajo y con una oferta hídrica estratégica para la población.

12.10. Conclusiones

Este capítulo presentó una caracterización ecohidrobiológica del humedal El Eneal centrada en la relación entre nutrientes, productividad primaria, estructura biológica y presencia de una planta carnívora acuática. El trabajo confirmó el primer registro de *Utricularia sp.* para este humedal y para los humedales de esta parte del golfo de Morrosquillo.

Los resultados mostraron que el sistema mantiene concentraciones bajas de nitrógeno y fósforo, coherentes con una condición oligotrófica, y que la productividad primaria bruta medida en 2022 fue inferior a la reportada para 2004. La coexistencia de *Najas guadalupensis* y *Utricularia sp.* en un contexto de baja biomasa planctónica y baja productividad sugiere que la estructura vegetal sumergida desempeña un papel ecológico decisivo en el funcionamiento del humedal.

Más allá del valor taxonómico del hallazgo, el capítulo muestra que registrar una especie puede ser también una puerta de entrada para reinterpretar el metabolismo ecológico de un sistema. En ese sentido, la presencia de *Utricularia sp.* no sólo amplía el inventario florístico de El Eneal: ayuda a comprender mejor la lógica interna de un humedal estratégico para la oferta hídrica regional.

Agradecimientos

Se agradece a la Universidad de Antioquia, al grupo GeoLimna y al proyecto sobre la evolución morfométrica e hidrobiológica de la ciénaga El Eneal por el soporte académico y científico que hizo posible este trabajo. Se reconoce también la participación de Adrián Humberto Escobar Vélez en el trabajo de campo y en la identificación ex post de las especies inventariadas, así como el apoyo de la reserva Sanguaré durante el desarrollo de las salidas de campo.

Referencias

- [1] E. L. Ríos Ortiz, F. d. J. Vélez Macías, D. S. Fernández Mc Cann, V. Naranjo Morales, M. G. Vallejo y N. J. Aguirre Ramírez, «Primer registro de *Utricularia* sp. en el humedal El Eneal, San Onofre, Colombia,» *Producción + Limpia*, vol. 18, n.º 2, 2023. DOI: 10.22507/pml.v18n2a4.
- [2] E. Ríos, J. Palacio y N. Aguirre, «Variabilidad fisicoquímica del agua en la ciénaga El Eneal, reserva natural Sanguaré, municipio de San Onofre, Sucre, Colombia,» *Revista Facultad de Ingeniería*, n.º 46, págs. 39-45, 2008.
- [3] E. Ríos, J. Palacio y N. Aguirre, «Productividad primaria y sustancias húmicas en la ciénaga El Eneal, San Onofre, Sucre, Colombia,» *Revista Facultad de Ingeniería*, n.º 47, págs. 67-72, 2009.
- [4] M. I. Críales-Hernández y M. Jerez-Guerrero, «Primer registro de *Utricularia foliosa* L. (Lentibulariaceae) y su espectro alimentario para la ciénaga de Paredes, Santander, Colombia,» *Actualidades Biológicas*, vol. 38, n.º 104, págs. 45-51, 2016.
- [5] C. Guisande et al., «Relative balance of the cost and benefit associated with carnivory in the tropical *Utricularia foliosa*,» *Aquatic Botany*, vol. 80, págs. 271-282, 2004.
- [6] C. Guisande, C. Granado-Lorencio, C. Andrade-Sossa y S. R. Duque, «Bladderworts,» *Functional Plant Science and Biotechnology*, vol. 1, págs. 58-68, 2007.
- [7] L. Sanabria-Aranda et al., «Predation by the tropical plant *Utricularia foliosa*,» *Freshwater Biology*, vol. 51, págs. 1999-2008, 2006.
- [8] T. Gaarder y H. Gran, «Investigations of the Production of Phytoplankton in the Oslo Fjord,» *Rapports et procès-verbaux des réunions*, vol. 42, 1927.
- [9] R. Margalef, *Limnología*. Omega, 1983.
- [10] N. Aguirre, *Hidrobiología sanitaria*. Universidad de Antioquia, 2013.

- [11] N. Aguirre y O. Caicedo, *Métodos de campo y de laboratorio para hidrobiología sanitaria*. Reimpresos Universidad de Antioquia, 2013.
- [12] A. Ramírez y G. Viña, *Limnología colombiana. Aportes a su conocimiento y estadísticas de análisis*. Fundación Universidad Jorge Tadeo Lozano, 1998.

Investigaciones 2023

Grupos de investigación GEOLIMNA y GEPAR



Este libro reúne una *selección de desarrollos, investigaciones y soluciones tecnológicas* construidas entre 2023 en la convergencia entre vision artificial, instrumentación, analítica de datos, automatización y *monitoreo ambiental*.

Los capítulos aquí reunidos muestran *cómo problemas reales del entorno pueden traducirse en sistemas medibles, algoritmos verificables, dispositivos funcionales y herramientas útiles* para la investigación, la *docencia y la transferencia de conocimiento*.

La obra refleja una forma de trabajo interdisciplinaria en la que la ingeniería dialoga con la limnología, la ecología, la agricultura de precisión, la infraestructura y el análisis ambiental. En ese cruce, los grupos GEOLIMNA y GEPAR *han consolidado una línea de producción académica y tecnológica orientada tanto a la comprensión de fenómenos como a la construcción de soluciones concretas*.

Universidad de Antioquia
Facultad de Ingeniería
Grupos de investigación GEOLIMNA y GEPAR
Medellin, Colombia
2026



Espacio reservado para ISBN, código de barras o sello editorial