

Proyecto: “Frailejon Detection”

David Hernando Avila Jimenez ^{a,c}, Erica Marcela Martínez Silva ^{a,c}, Lina Marcela Rivas^{a,c},
Raúl Eduardo Vásquez Duarte ^{a,c}, Sebastián Amaya Porras^{a,c}

Sergio Alberto Mora Pardo^{b,c}

^a*Estudiantes de la Maestría en Analítica para Inteligencia de negocios*

^b*Profesor, Departamento de Ingeniería Industrial*

^c*Pontificia Universidad Javeriana, Bogotá, Colombia*

1. ENTENDIMIENTO DE NEGOCIO

1.1 Contexto de negocio.

Los páramos de Chingaza, Cruz Verde y Sumapaz representan algunos de los ecosistemas más valiosos y frágiles de Colombia, situados en las altas montañas de los Andes a altitudes que superan los 3,000 metros sobre el nivel del mar. Estos ecosistemas desempeñan un papel crucial en la regulación hídrica, actuando como gigantescas esponjas que capturan y liberan agua lentamente, garantizando el abastecimiento hídrico a millones de personas en las regiones circundantes.

El frailejón, perteneciente al género *Espeletia*, es una planta emblemática de estos páramos y juega un rol fundamental en el mantenimiento del equilibrio ecológico. Estas plantas absorben la humedad de la niebla a través de sus hojas y la dirigen al suelo con sus tallos, contribuyendo a regular el ciclo del agua. Su capacidad de adaptación a las duras condiciones climáticas de los páramos, incluyendo temperaturas bajas y alta radiación ultravioleta, las convierte en especies clave para la estabilidad del ecosistema.

Además de su función hídrica, los páramos son reservorios de biodiversidad y secuestran grandes cantidades de carbono en sus suelos, ayudando a mitigar el cambio climático. Sin embargo, estos ecosistemas enfrentan graves amenazas debido a actividades humanas como la expansión agrícola, la minería y el pastoreo, así como los efectos del cambio climático, lo que pone en peligro tanto a los páramos como a los frailejones. Por ello, la conservación y el manejo sostenible de estos ecosistemas son esenciales para garantizar la supervivencia de las especies que los habitan y la disponibilidad de recursos vitales para las comunidades humanas.

Uno de los grandes desafíos nacionales es la conservación y mapeo de los frailejones en los páramos, para proteger las áreas donde se encuentran y preservar los recursos naturales. Por lo tanto, se han implementado tecnologías avanzadas, como el uso de fotografía satelital, con el apoyo del procesamiento de datos mediante modelos de aprendizaje profundo (Deep Learning). Estas tecnologías permiten la identificación de frailejones a través del reconocimiento de imágenes, facilitando la conservación de estos ecosistemas.

La combinación de imágenes satelitales y modelos de aprendizaje profundo no solo mejora la detección y el monitoreo de los frailejones, sino que también permite la creación de mapas detallados de su distribución y densidad. Este enfoque proporciona una base sólida para la toma de decisiones informadas en la gestión y protección de los páramos, asegurando la preservación a largo plazo de estos recursos naturales y contribuyendo a la sostenibilidad ambiental.

Para el entendimiento a nivel interno y externo del sector se llevó el estudio de 2 metodologías diferentes PESTEL y DOFA.



Ilustración 1 Matriz DOFA.
Fuente: Elaboración propia basado en (Rojas Rojas et al., 2018)



Ilustración 2. Análisis PESTEL. Elaboración propia

1.1.1. Objetivo de negocio:

- Contribuir al proyecto de Ingeniería enfocado en la preservación del ecosistema del páramo, mediante la identificación automática de zonas con baja densidad de Espeletia.
- Proveer una posible solución que facilite la monitorización del ecosistema del páramo, apoyando la conservación de su potencial hídrico.

1.1.2. Criterio de éxito:

- Un análisis cualitativo de la imagen IMG_3451.JPG, donde muestre los Frailejones detectados en el área del páramo tomado por la fotografía; para identificar zonas de baja y alta densidad de frailejones.

1.2. Determinación de objetivos de minería de datos

1.2.1. Objetivo minería de datos

- Desarrollar 6 modelos de redes neuronales capaces de clasificar imágenes aéreas y que permitan la identificación de frailejones en los páramos de Chingaza y cruz verde.

1.2.2. Criterio de éxito

- Obtener, para cada modelo desarrollado, los siguientes resultados en la métrica de “AUC”.

Modelo	Resultado esperado
Sencilla	0.88
Multicapa	0.95
Convolutacional	0.95
Convolutacional+VGG16	0.99
Convolutacional+VGG16(Optuna)	0.99
Adicional, según escogencia de los analistas.	0.96

2. ENTENDIMIENTO DE LOS DATOS

Para el proyecto se tomarán dos conjuntos de datos:

1. Archivo “Data_F”, el cual se usará para realizar el entrenamiento de los modelos. Este archivo es de tipo **HDF5** (Hierarchical Data Format version 5), formato de archivo que permite almacenar y organizar grandes volúmenes de datos complejos.

Para este caso, se tiene un dataset de 21 imágenes almacenadas, con una dimensión de 100x100 píxeles cada una, y con un espacio de color RGB.

2. Imagen “IMG_3451.JPG”, para realizar el análisis cualitativo en donde se muestre los Frilejones detectados en el área del páramo. Esta imagen tiene las siguientes características:
 - a. Dimensiones 4000 x 3000
 - b. Espacio de Color: RGB
 - c. Longitud focal:5,2
 - d. Canal alfa: No
 - e. Ojos rojos: No
 - f. Modalidad de medición: patrón
 - g. Número f: f/4
 - h. Programa de exposición: Normal
 - i. Tiempo de exposición: 1/1.000
 - j. Latitud: 4° 34' 26,79” N
 - k. Longitud: 74° 1' 22,859” O

3. PREPARACIÓN DE LOS DATOS

3.1 Limpieza de datos

Para la limpieza de los datos, se utilizó una función llamada **Import Images**, creada por el profesor Camilo Franco Ph.D; en la cual realiza la extracción, procesamiento y preparación del archivo de entrenamiento Data_F. En esta función, se aplanan las imágenes para convertirlas en vectores columna, se normalizan y

con ello se obtiene el resultado en arreglos; con esto estarían listos para usar en los modelos que se desarrollarán.




3.2 Transformación de los datos

Posteriormente, se realizó trasposición en las variables que resultaron de la función **Import Images**, con el fin de obtener correctamente (muestras, características) y con ello desarrollar los modelos debidamente.

Así mismo, se realizaron redimensionamientos de acuerdo con la necesidad del modelo; por ejemplo, para el modelo de ResNet las imágenes debían tener un tamaño de 70 x 70 y para poder realizar el modelo de VGG, este, debe recibir un formato con tamaño (224 x 224).

4. MODELING & EVALUATION

Para el proyecto, se desarrollaron 6 modelos los cuales se describen a continuación:

Modelo	Parámetros	Resultado	Notebook
Red neuronal sencilla	Número de neuronas en la capa oculta: 428 Número de épocas: 163 Tasa de aprendizaje: 3.9563664553994944e-05 Tasa de dropout: 0.2948388655655226 Tamaño de batch: 325 Mejor época: 163 (indicado por el EarlyStopping)	Loss: 0.4383 AUC: 0.883	
Red neuronal multicapa	2 Capas Número de neuronas en la primera capa oculta: 159 Número de neuronas en la segunda capa oculta: 457 Número de épocas: 500 Tasa de aprendizaje: 0.0007375043276204938 Tasa de dropout: 0.3166866137483689 Tamaño de batch: 130 Mejor época: 500 (por el EarlyStopping se ha detenido en la mejor época)	Loss: 1.2703 AUC: 0.9482	
Red convolucional	<ul style="list-style-type: none"> 4 capas de convolución, cada una seguida de una normalización por lotes y max-pooling. Función de activación “Relu” El tamaño del kernel es de 3x3, el método de inicialización es el método “he” después de las capas se utiliza un “ Flatten” para el aplanado de las salidas y se conectan con dos capas densas que incluyen una regularización l2 y un Dropout esto con el fin de reducir del sobreajuste. La Capa de salida es una neurona con activación sigmoide ya que es un problema de clasificación binaria. Optimizador: “Adam”. Learning rate: 0.00159. funcion de Perdida: 'binary_crossentropy'. epochs= 100 batch_size= 8 Mejores hiperparámetros (optuna): filters_1: 118. filters_2: 84. filters_3: 156.	AUC: 0.98 Precisión: 0.92 Loss: 0.2	


	filters_4: 146. dense_units_1: 123 dense_units_2: 37 dropout_rate: 0.5079153648495992 learning_rate: 0.00159039859750974		
RestNet	<ul style="list-style-type: none"> • weights='imagenet': El modelo preentrenado ResNet-50 utiliza los pesos preentrenados en ImageNet. • GlobalAveragePooling2D: Aplica un promedio global para reducir la dimensionalidad de las características extraídas por ResNet-50. • Dense(2048, activation='relu') • Dropout(0.3): Implementa Dropout con una tasa de abandono del 30% para prevenir sobreajuste. • Dense(1024, activation='relu') • Dropout(0.3) • Dense(1, activation='sigmoid') • kernel_regularizer=tf.keras.regularizers.l2(0.01): Se aplica regularización L2 en las capas densas para prevenir sobreajuste. • Se descongelaron las últimas 10 capas del modelo ResNet-50 para permitir su ajuste durante el entrenamiento. • Adam(learning_rate=1e-5): Se utiliza el optimizador Adam con una tasa de aprendizaje baja (1e-5) para el fine-tuning de las capas descongeladas. • epochs=250 • batch_size=8 	Precisión: 0.89 AUC: 0.96	
Red convolucional con transferencia VGG-16	Modelo Base: VGG. Capas adicionales al modelo base: 6 <ul style="list-style-type: none"> • 1 capa de global Average Pooling. • 2 capas Dense (512 & 256 neuronas, ambas con función de activación Relu) • 2 capas de Dropout (0.5) 	AUC: 0.9902 Loss: 0.099	
Red convolucional con transferencia VGG-16 + Ajuste de hiperparámetros	Misma estructura que la línea anterior, se le agregó el uso de Optuna , a partir del cual se obtuvo: learning_rate: 0.00072 units(Neuronas en la capa): 128 dropout_rate: 0.229	AUC: 0.9945 Loss: 0.04	

Tabla 1.Descripción de modelos realizados con sus respectivos parámetros, resultados y link de GitHub donde se puede encontrar el código.

4.1 Asses Model

De acuerdo con la tabla presentada, se detalla la construcción de los modelos, destacando su comportamiento y capacidad de clasificación, evaluados mediante el AUC, la métrica clave utilizada para medir el éxito de los modelos desarrollados.

4.1.1 Red Sencilla

El modelo de red neuronal sencilla se configuró con una única capa oculta que cuenta con 428 neuronas, entrenado durante 163 épocas. Se utilizó una tasa de aprendizaje muy baja de 3.9563664553994944e-05,

junto con una tasa de dropout del 29.48% para prevenir el sobreajuste. El tamaño de batch fue de 325. El entrenamiento se detuvo automáticamente en la época 163 gracias a la técnica de EarlyStopping. Al evaluar el modelo, se obtuvo una pérdida (loss) de 0.4383 y un AUC de 0.883.



Ilustración 4 Predicción en IMG_3451.JPG usando Red neuronal sencilla (Elaboración propia)

4.1.2 Red Multicapa

Se configuró con dos capas ocultas. La primera capa tiene 159 neuronas, mientras que la segunda tiene 457 neuronas. El modelo fue entrenado durante 500 épocas, usando una tasa de aprendizaje de 0.0007375043276204938 y una tasa de dropout de 0.3166866137483689 para prevenir el sobreajuste. El tamaño de batch fue de 130. El entrenamiento se detuvo en la mejor época gracias al uso de EarlyStopping. Los resultados del modelo incluyen una loss en validación de 1.2703 y un AUC en validación de 0.9482, lo que sugiere un buen rendimiento del modelo en términos de capacidad de clasificación, a pesar de una pérdida relativamente alta.

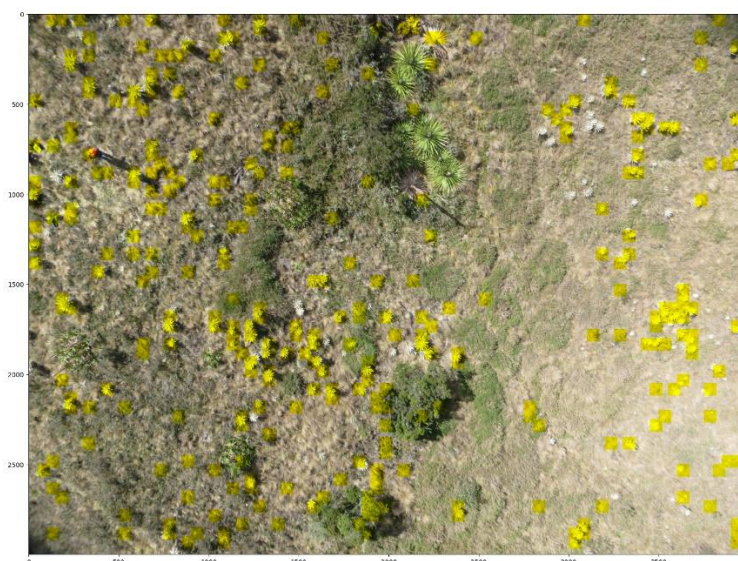


Ilustración 4 Predicción en IMG_3451.JPG usando Red neuronal multicapa (Elaboración propia)

4.1.3 Red Convolucional

En este modelo, se desarrollaron cuatro capas de convolución bidimensionales con el objetivo de extraer diversas características de las imágenes, utilizando los siguientes parámetros:

- **kernel_size=3**: Se seleccionó un filtro de 3 x 3 píxeles para capturar detalles finos en las imágenes.
- **padding="same"**: Se utilizó padding para asegurar que todos los bordes de la imagen sean considerados al aplicar los filtros, manteniendo el tamaño original de la imagen.
- **activation="relu"**: Se aplicó la función de activación ReLU, que establece la salida en cero para entradas negativas y la iguala a la entrada cuando ésta es positiva.
- **kernel_initializer="he_normal"**: Los pesos de los filtros se inicializaron utilizando la técnica He, optimizada para capas con activación ReLU.

Estos parámetros fueron ajustados mediante Optuna, con el objetivo de optimizar el rendimiento del modelo y alcanzar la meta establecida de una precisión de 0.95. Gracias a este proceso de optimización, se lograron identificar los mejores hiperparámetros, permitiendo al modelo alcanzar un AUC de 0.98, superando la expectativa inicial.

El resultado al aplicar la red convolucional hacia el análisis cualitativo se obtuvo lo siguiente:

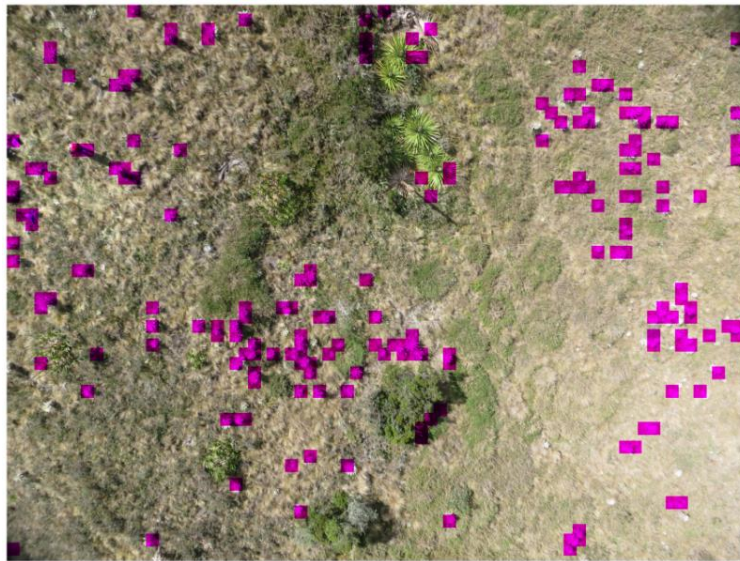


Ilustración 5 Predicción en IMG_3451.JPG usando Red Convolucional (Elaboración propia)

4.1.4 Red Convolucional con transferencia VGG16

Este modelo usó una base pre entrenada con la arquitectura VGG-16 y a partir de allí, se finalizó con una red neuronal sencilla como modelo completo para la predicción de los frailejones. Para esta ocasión se estipularon los siguientes parámetros:

- **Modelo Base**: VGG16, con pesos alineados a imagenet.
- Se descongelaron o usaron 4 capas del modelo base, para que éste pudiese aprender de los datos del modelo actual y así tener un mejor resultado.
- La arquitectura Secuencial: con las capas mencionadas en la tabla 1., se decidió mantener las funciones de activación RELU y sigmoide para la última

El AUC aumentó en 0,01, un valor no muy diferenciador frente a la red convolucional desarrollada anteriormente; pero el aumento puede reflejarse por la transferencia de información del modelo base, genera una ganancia potencial; además, el tiempo que se desarrolla demuestra la mejoría en el rendimiento del modelo. El resultado de este modelo, al revisar el análisis cualitativo sobre la imagen, fue el siguiente:

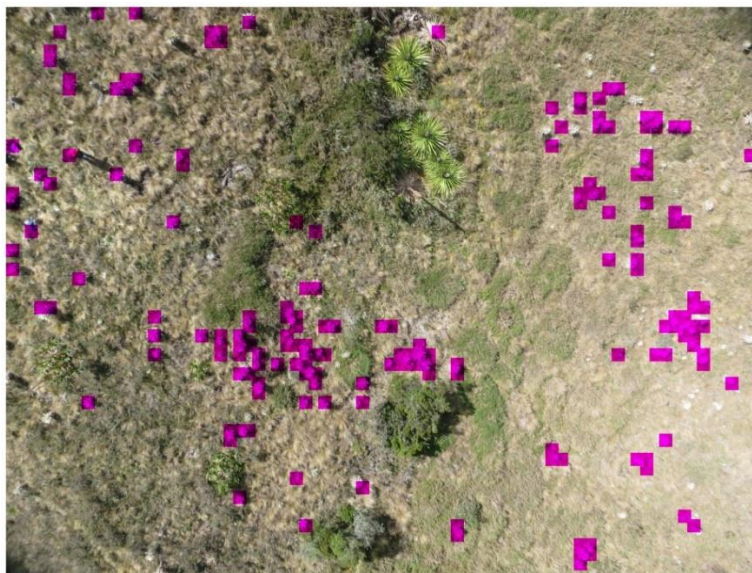


Ilustración 6 Predicción en IMG_3451.JPG usando Red Convolucional con transferencia VGG16 (Elaboración propia)

4.1.5 Red Convolucional con transferencia VGG16 y optimización de hiperparámetros

Finalmente, se aplicó el framework de optimización de hiperparámetros Optuna, para verificar los mejores valores sobre dicho modelo. Alcanzando altos valores en el AUC, como parámetro de comparación. Se observa una mejoría en cuando a puntos porcentuales sobre el AUC, debido a la transferencia de conocimiento del modelo base, reflejado en predicciones más precisas sobre la imagen.

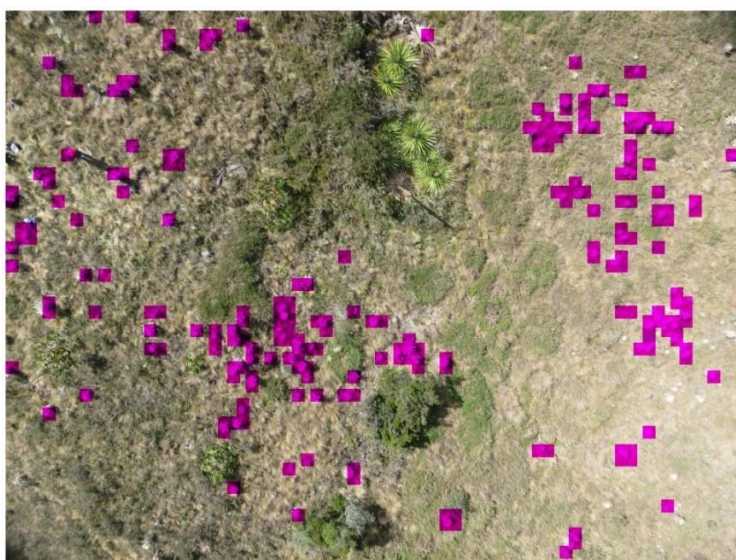



Ilustración 7 Predicción en IMG_3451.JPG usando Red Convolucional con transferencia VGG16 y optimización (Elaboración propia)

5. ANÁLISIS CUALITATIVO

Para realizar el análisis cualitativo, se llevó a cabo sobre el modelo que arrojó el mejor resultado de AUC (VGG16) y se utilizaron dos métodos descritos a continuación, los cuales se pueden ver a detalle en el

siguiente link: 

1. **Método manual para densidad general:** se calcularon las predicciones realizadas por el modelo y se dividieron entre el tamaño de la imagen el cual era de 4000x3000 píxeles, resultando $2e^{-5}$ frailejones/píxeles² (227/12MM frailejones/píxeles²) de densidad total sobre la imagen.

2. **Usando GMM:** Se utilizó la técnica econométrica de estimación de parámetros para obtener la densidad poblacional por clases o grupos. Dado que en el conjunto de datos que se trabajan en el presente proyecto no hay clases o tipos de frailejones, se utilizó la técnica de aprendizaje no supervisado PCA (Principal Components Analysis), para generar la mejor agrupación de los datos, en este caso, los frailejones. A partir de allí, se obtuvo que lo mejor sería subdividirlo en 4 grupos o componententes (Gráfica de la izquierda en la Ilustración 3. mostrada a continuación) y posteriormente, aplicando GMM se obtienen las densidades correspondientes a los 4 grupos generados, como se muestra en la gráfica de la izquierda.

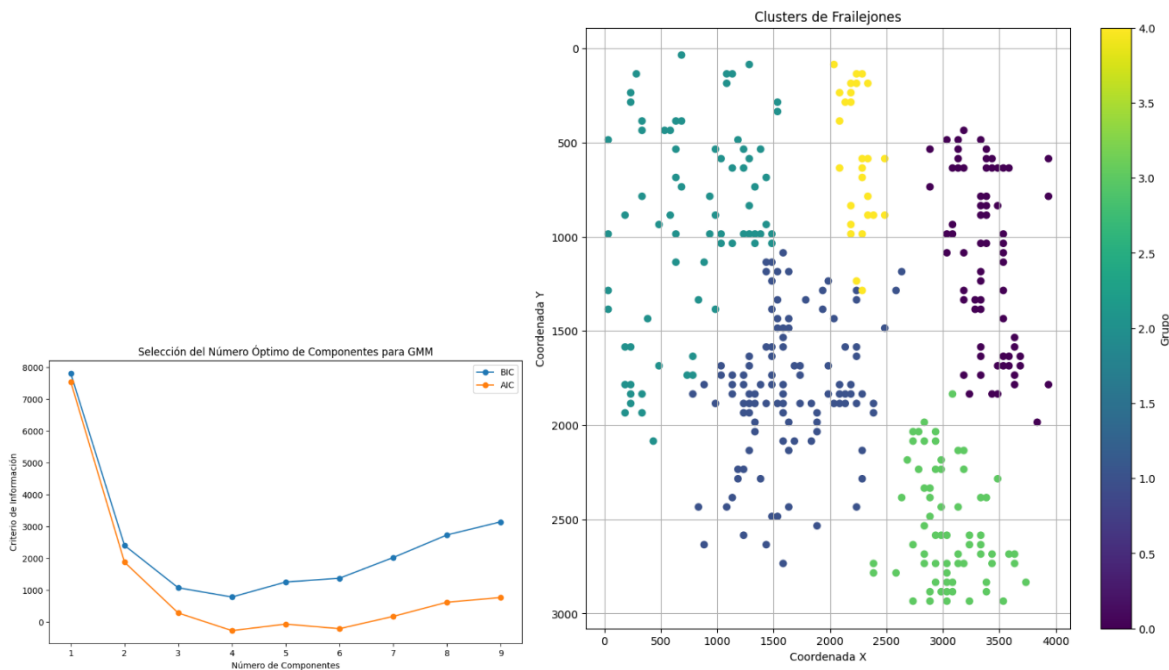


Ilustración 3. (izq.) Componentes principales para definición de grupos a los que se le realiza el cálculo de la densidad. (dere.) gráfico de dispersión y densidad poblacional por grupos armados.

La densidad por grupo fue la siguiente:

GRUPO	DENSIDAD (Frailejones/píxeles ²)
1	0.000047
2	0.000049
3	0.000032
4	0.000056

Tabla 2 Descripción de densidad poblaciones de los frailejones

Las medidas pueden ser con valores muy pequeños debido al tamaño de la imagen, la cantidad de píxeles es de 12'000.0000, lo cual conlleva a reducir la magnitud de la densidad.

6. DEPLOYMENT

A partir de los objetivos planteados, de modelos desarrollados y de los resultados obtenidos, se propone una serie de estrategias enfocada en el despliegue y pasos a seguir a fin de fortalecer la preservación del ecosistema de los páramos e identificar frailejones mediante modelos de redes neuronales.

Estrategias:

- Formar consorcios multidisciplinarios que involucren ingenieros, biólogos y ecólogos de la mano del apoyo de comunidades locales, ONGs y entidades gubernamentales, en busca de garantizar una aproximación integral al monitoreo del ecosistema, con aportes desde la ciencia, la tecnología y el conocimiento local.
- Incentivar la participación de investigadores y desarrolladores mediante hackathons o competencias organizadas en plataformas como Kaggle para optimizar el rendimiento de los modelos.
- Implementar plataformas para almacenamiento y análisis de las imágenes aéreas de los páramos, a fin de lograr eficiencia en los procesos de almacenar, procesar y compartir información con facilidad, facilitando la colaboración interinstitucional.
- Desarrollar planes piloto con los seis modelos de redes neuronales en áreas clave como Chingaza y Cruz Verde, para posteriormente escalar la solución a otros páramos en Colombia.

Potenciales siguientes pasos de la investigación

Paso	Descripción
Validación y Refinamiento	Revisar y ajustar el modelo, optimizando hiperparámetros y mejorando la precisión mediante técnicas avanzadas de regularización y aumento de datos.
Postprocesamiento de Resultados	Refinar predicciones con técnicas adicionales como filtrado de ruido y clustering para mejorar la estimación de densidad.
Análisis Geoespacial	Integrar resultados en sistemas de información geográfica (SIG) para un análisis espacial detallado y generar mapas de distribución y patrones espaciales.
Monitoreo y Seguimiento	Implementar un sistema de monitoreo continuo para observar cambios en la densidad de frailejones a lo largo del tiempo.
Integración con Modelos Ecológicos	Incorporar los resultados en modelos ecológicos que simulen el impacto de factores ambientales en la población de frailejones.
Publicación y Compartición	Publicar los hallazgos en revistas científicas y compartir datos y modelos con la comunidad para fomentar la colaboración y la toma de decisiones.
Desarrollo de Aplicaciones	Desarrollar herramientas prácticas o aplicaciones para el monitoreo y protección de frailejones, como plataformas de monitoreo en tiempo real.

Tabla 3 Descripción de los pasos potenciales a seguir para la investigación