

# ANÁLISIS DE INFORMACIÓN DE IMÁGENES SATELITALES APOYADA CON PROGRAMACIÓN EN PYTHON: APLICACIÓN A LA GEORREFERENCIACIÓN EN INGENIERÍA

## Resumen

Se presenta en este documento un trabajo relacionado con la extracción y análisis de información de imágenes satelitales, apoyada en la programación con Python. En este escenario, se abordó un estudio de caso ubicado en el Departamento de Cundinamarca Colombia, inicialmente, se seleccionaron ciertas capas de la región para desarrollar algunos scripts y se presentan los resultados del potencial de usar la programación para acciones repetitivas en lugar de hacerlo manualmente.

Es importante anotar que este trabajo, se constituye en un referente de desarrollo en el área de programación ya que permite usar el lenguaje Python para crear scripts en apoyo al trabajo de campo en ingeniería.

**Keywords-** Qgis, Layers, Python, PyQgis, scripts.

## I. INTRODUCCIÓN

En el desarrollo del proyecto de investigación sobre la calidad del agua en el municipio de Zipaquirá, Cundinamarca (Col.) se han realizado levantamientos topográficos y mapas georreferenciados usando localización con GPS y fotografía con drones, estos mapas se contrastaron con los mapas que tienen instituciones como el Geoportal IGAC y la plataforma Colombia en mapas, estos mapas son seleccionados de acuerdo a las layers que contienen información pertinentes para la investigación en curso, los mapas se están descargando en formato SHP (Shape) y se utiliza la aplicación Qgis para crear los mapas georreferenciados.

## II. MATERIALES Y MÉTODOS

En el marco del uso de herramientas de programación en sistemas GIS se desarrolló la siguiente metodología de trabajo:

### Ubicación de la zona de estudio

El presente artículo también pretende aparte de los aportes al proyecto de investigación y al tema de georreferenciación dar un modelo básico paso a paso, para trabajar con las aplicaciones Qgis (georreferenciación) y Python (Lenguaje de

programación), ambas de software libre para permitir el acceso de cualquier usuario sin incurrir en gastos de licenciamiento.

Iniciamos cargando un mapa satelital y ubicamos la zona de estudio, los hacemos agregando XYZ Tiles una nueva conexión, en la siguiente tabla se muestran las conexiones con google:

TABLA I  
ENLACES PARA CREAR LAS CONEXIONES EN QGIS.

Nombre de la capa	Dirección para instalar la capa
Google Maps	<a href="https://mt1.google.com/vt/lyrs=r&amp;x={x}&amp;y={y}&amp;z={z}">https://mt1.google.com/vt/lyrs=r&amp;x={x}&amp;y={y}&amp;z={z}</a>
Google Satellite	<a href="http://www.google.cn/maps/vt?lyrs=s@189&amp;gl=cn&amp;x={x}&amp;y={y}&amp;z={z}">http://www.google.cn/maps/vt?lyrs=s@189&amp;gl=cn&amp;x={x}&amp;y={y}&amp;z={z}</a>
Google Hybrid	<a href="https://mt1.google.com/vt/lyrs=y&amp;x={x}&amp;y={y}&amp;z={z}">https://mt1.google.com/vt/lyrs=y&amp;x={x}&amp;y={y}&amp;z={z}</a>
Google Terrain	<a href="https://mt1.google.com/vt/lyrs=t&amp;x={x}&amp;y={y}&amp;z={z}">https://mt1.google.com/vt/lyrs=t&amp;x={x}&amp;y={y}&amp;z={z}</a>
Google Traffic	<a href="https://mt1.google.com/vt?lyrs=h@15900000,traffic seconds_into_week:-1&amp;style=3&amp;x={x}&amp;y={y}&amp;z={z}">https://mt1.google.com/vt?lyrs=h@15900000,traffic seconds_into_week:-1&amp;style=3&amp;x={x}&amp;y={y}&amp;z={z}</a>
Google Roads	<a href="https://mt1.google.com/vt/lyrs=h&amp;x={x}&amp;y={y}&amp;z={z}">https://mt1.google.com/vt/lyrs=h&amp;x={x}&amp;y={y}&amp;z={z}</a>

También podemos agregar la capa Bing Satellite que junto con Google Satellite son de las más utilizadas.

Bing Satellite:  
[http://ecn.t3.tiles.virtualearth.net/...{q}.jpeg?g=0&dir=dir\\_n](http://ecn.t3.tiles.virtualearth.net/...{q}.jpeg?g=0&dir=dir_n)

### Búsqueda y carga de las capas requeridas.

Una vez tenemos activa alguna de estas capas o layers de vista satelital, buscamos en los archivos .shp descargados de fuentes

confiables del (los) archivo(s) que necesitemos, por ejemplo archivo de red fluvial, red de carreteras, reservas de agua, área de bosque, zonas de explotación minera, división política y muchas más opciones que se encuentran en estos archivos, y podemos superponer sobre la capa satelital la capa hidrografía.

### Uso de la API PyQgis para extraer información de los mapas.

QGIS es un Sistema de Información Geográfica (SIG) de Código Abierto licenciado bajo GNU - General Public License. QGIS es un proyecto oficial de Open Source Geospatial Foundation (OSGeo). Corre sobre Linux, Unix, Mac OSX, Windows y Android y soporta numerosos formatos y funcionalidades de datos vector, datos ráster y bases de datos. [1].

Python es un lenguaje de programación potente y fácil de aprender. Tiene estructuras de datos de alto nivel eficientes y un simple pero efectivo sistema de programación orientado a objetos. La elegante sintaxis de Python y su tipado (tipos de datos de las variables) dinámico, junto a su naturaleza interpretada lo convierten en un lenguaje ideal para scripting y desarrollo rápido de aplicaciones en muchas áreas, para la mayoría de plataformas. [2]

El uso de la API (interfaz de programación de aplicaciones) Pyqgis permite al usuario crear scripts o códigos de programación para tareas repetitivas y manuales que se realizan con Qgis, unir capas, determinar distancias y áreas, reproyectar datasets, lo que permite ahorrar tiempo y disminuir los errores al realizar acciones manualmente.

Permite desarrollar herramientas personalizadas para análisis que no están disponibles en Qgis, con los scripts generados se pueden modificar capas, mapas y etiquetas de forma dinámica, y otras ventajas como:

Acceso a las librerías de Python para realizar machine learning, análisis estadístico y manejo de datos.

Portabilidad: Los scripts creados funcionan como aplicaciones Python independiente de Qgis, por lo se pueden usar con otras aplicaciones.

Costo: Las aplicaciones Qgis y Python son software libre, y tienen el apoyo de la comunidad de programadores y documentación en línea.

En resumen el uso Qgis + Python permite versatilidad y potenciar las tareas y los análisis cuando se requiere más que las funciones básicas propias de Qgis.

Para iniciar el uso de la interfaz con Python se deben cargar algunos módulos o librerías.

Para cargar un proyecto se requiere de las siguientes instrucciones:

Se QgsVectorLayer crea una instancia especificando el nombre de un proveedor de datos, como postgres o wfs, y la URL que define el conjunto de datos específico al que conectarse. El constructor de la capa vectorial, a su vez, crea una instancia de una QgsVectorDataProvidersubclase correspondiente al tipo de proveedor y le pasa la URL. El proveedor de datos se conecta a la fuente de datos.[3]

```
from qgis.core import QgsVectorLayer
```

```
vlayer = QgsVectorLayer("d:/AFG/2023/proyecto de investigacion/laccei2024/HidrografADa._MunicipioZipa...
```

Una vez cargado el proyecto y las capas necesarias podemos empezar a diseñar los scripts para hacer consultas.

Las capas tienen definidos unos atributos que pueden ser utilizados por los programadores para filtrar o depurar información de las capas. Lo descrito se representa en la figura 1.

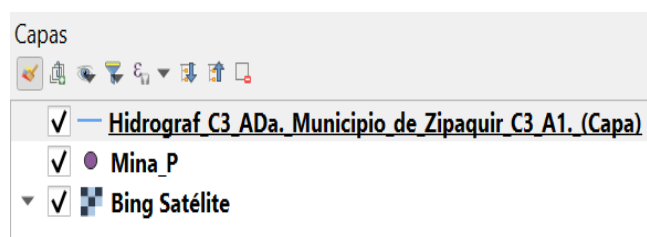


Fig. 1 Capas activas en Qgis.  
Fuente: Elaboración propia

En el entorno de Qgis podemos tener varias capas y activar o desactivar las que necesitemos, o superponer una sobre otra como lo presentamos en el presente trabajo.

Estas capas se pueden configurar para una mejor visualización, cambiar fuente de las etiquetas, los colores y grosor de líneas y demás objetos que aparecen en una capa.

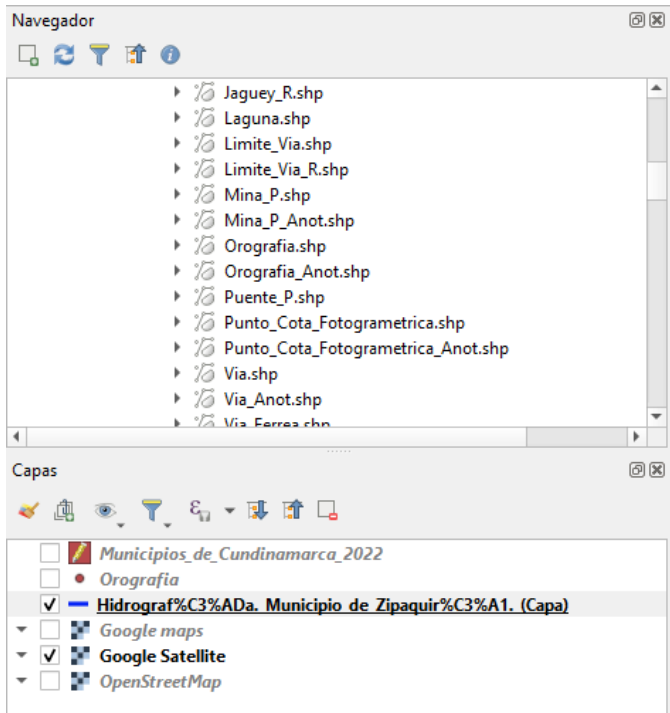


Fig. 2 Capas instaladas, activas e inactivas.  
Fuente: Elaboración propia

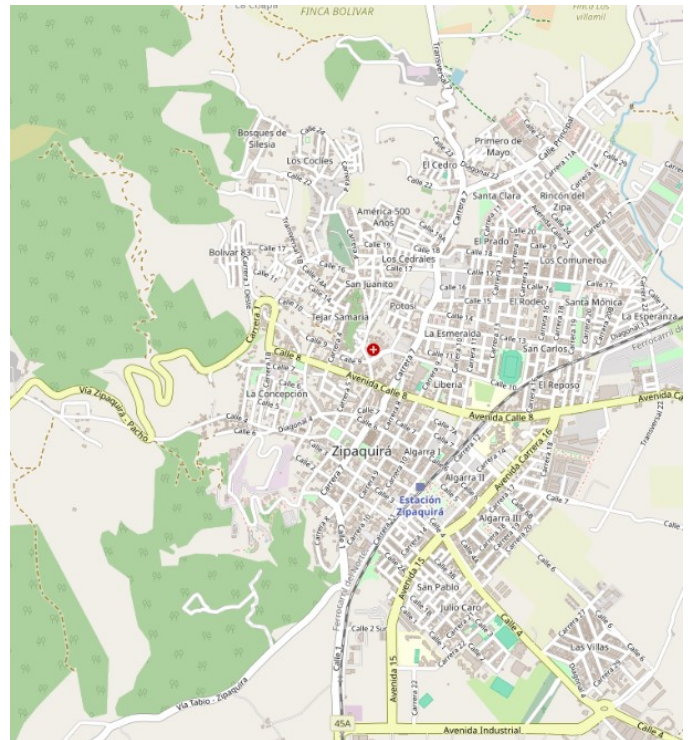


Fig. 3 Área de estudio, Municipio de Zipaquirá.  
Fuente: Elaboración propia

Una vez seleccionada el área de trabajo, en este caso el municipio de Zipaquirá, agregamos las capas de Hidrografía y de minas de carbón del área de estudio. Como se ve en la figura 2

Así podemos visualizar distancias entre fuentes de agua y las minas de carbón, conociendo los efectos contaminantes de la práctica de la minería y teniendo en cuenta los atributos de las capas de hidrografía y minería para empezar a buscar distancias entre algunos puntos, esto se puede lograr con programación en Python usando la API Pyqgis.

### III RESULTADOS

#### Ubicación de la zona de estudio

Usando las capas de vista satelital como Google y Bing se delimita el área de trabajo al Municipio de Zipaquirá Cundinamarca, donde se puede observar la zona urbana y la zona rural.

A continuación se muestra la zona de estudio en la figura 3

#### Búsqueda y carga de las capas requeridas.



Fig. 4 Hidrografía del Municipio de Zipaquirá  
Fuente: <https://www.colombianmaps.gov.co/>



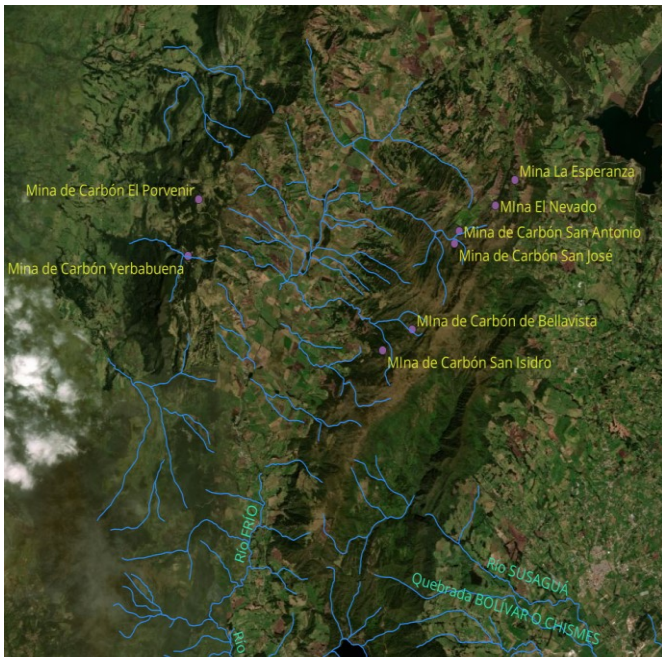


Fig. 5 Vista de minas de carbón cercanas a fuentes de agua en Zipaquirá Cundinamarca

Fuente: <https://www.colombianmaps.gov.co/>



Fig. 6 Ubicación de la Mina de carbón yerbabuena

Fuente: Elaboración propia.

Usando la interfaz de Pyqgis se realizaron consultas sobre una de las capas utilizadas en la georreferenciación, en la tabla a continuación se muestran algunos de los atributos:

TABLA III  
ATRIBUTOS DE LA CAPA HIDROGRAFÍA EN QGIS

OBJECTID_1	OBJECTID	NOMBRE	SHAPE_Leng	Shape_Len
94	94	103 NULL	1090,967828000000054	1090,967789161069959
95	95	104 NULL	625,47734969900001	625,477345184903015
96	96	105 Quebrada DEL GUERRERO	4725,543314509999618	4725,543119860620209
97	97	106 NULL	896,548205825999958	896,548212612672046
98	98	107 NULL	1247,698457720000079	1247,698511077290050
99	99	108 NULL	734,178994794000005	734,178964562887018
100	100	109 NULL	709,747828888999948	709,747855568397995
101	101	110 NULL	1046,596727009999995	1046,596719399919948
102	102	575 Quebrada EL AMOLADERO	6307,627075880000120	6307,627524994830310
103	103	645 Río SUSAGUÁ	11711,9319709999999	11711,931721355200352
104	104	749 NULL	1291,775692070000105	1291,775800348430039
105	105	111 NULL	1177,008969889999989	1177,008952992149943

A continuación esta tabla nos muestra los datos de las minas de carbón ubicadas en la zona.

TABLA II  
LISTA DE LAS MINAS DE CARBÓN EN LA ZONA DE ESTUDIO

NOMBRE_GEO	SYMBOL	PROYECTO
1 Mina de Carbón San Isidro	1	NULL
2 Mina de Carbón de Bellavista	1	NULL
3 Mina de Carbón Yerbabuena	1	NULL
4 Mina de Carbón San José	1	NULL
5 Mina de Carbón San Antonio	1	NULL
6 Mina El Nevado	1	NULL
7 Mina de Carbón El Porvenir	1	NULL
8 Mina La Esperanza	1	NULL

Vamos a seleccionar la mina de carbón Yerbabuena y buscar su ubicación específica en el mapa ver figura 6.

### Uso de la API PyQgis para extraer información de los mapas.

Usando los comandos de Python en la interfaz de PyQgis realizamos consultas para corroborar el funcionamiento del script comparando con los datos que nos muestran los atributos de la tabla.

Ejemplo en la figura 8 nos muestran un total de 305 objetos identificados en esta capa.

Con el siguiente script podemos obtener la información del número de objetos de la capa:

```

1 from qgis.core import QgsVectorLayer
2 vlayer = QgsVectorLayer("d:/AFG/2023/proyecto de investigacion/laccei 2024/HidrografAda
3 nfeat = vlayer.featureCount()
4 print(nfeat)

```

Fig. 7 Comandos para crear una capa y mostrar el número de objetos.  
Fuente: Elaboración propia

Al ejecutar el anterior script la ventana de ejecución muestra la siguiente salida:

```

r.more.info
3 # Security warning: typing commands from an untrusted source can harm your computer
4 >>> exec(Path('C:/Users/ARIELG~1/AppData/Local/Temp/tmpjwemmu11').read_text())
5 305
6

```

Fig. 8 Consola de Python mostrando resultados de una consulta sobre número de objetos en una capa.  
Fuente: Elaboración propia

En figura 10 se muestran los campos que componen la tabla de atributos, para conseguir la información de los nombres de los campos y el tipo de campo diseñamos el siguiente script:

```

1 from qgis.core import QgsVectorLayer
2 vlayer = QgsVectorLayer("d:/AFG/2023/proyecto de investigaci
3 campos = vlayer.fields()
4 campos
5 for campo in campos:
6     print(campo.name(), campo.typeName())
7

```

Fig. 9 Script para conseguir el nombre y tipo de los campos de una capa  
Fuente: Elaboración propia

```

1 # Python Console
2 # Use iface to access QGIS API interface or type '?' for more info
3 # Security warning: typing commands from an untrusted source can harm your computer
4 >>> exec(Path('C:/Users/ARIELG~1/AppData/Local/Temp/tmp61e99kkm').read_text())
5 OBJECTID Integer
6 OBJECTID Integer
7 NOMBRE String
8 SHAPE_Leng Real
9 Shape__Len Real
10

```

Fig. 10 Resultados de la consulta nombre y tipo de campos de una capa.  
Fuente: Elaboración propia

También podemos ejecutar las instrucciones directamente desde la consola de Python como se muestra en el siguiente ejemplo:

```

1 # Python Console
2 # Use iface to access QGIS API interface or type '?' for more info
3 # Security warning: typing commands from an untrusted source can harm your computer
4 >>> rect = vlayer.extent()
5 >>> rect
6 <QgsRectangle: -74.10243648344750511 4.96147883282015023, -73.90798217521779634 5.16631890720849984>
7 >>> rect.width()
8 0.19445430822970877
9 >>> rect.height()
10 0.2048400743883496
11 >>> rect.xMaximum()
12 -73.9079821752178
13 >>> rect.yMaximum()
14 5.1663189072085
15

```

Fig. 11 Obtener información con comandos predeterminados, como las coordenadas, ancho, alto, coordenada máxima en X y Y.  
Fuente: Elaboración propia

Como se observa con los comandos del script anterior figura 11, podemos obtener información sobre coordenadas (QgsRectangle), ancho (rect.width), alto (rect.height), coordenada máxima en x (rect.xMaximum) y coordenada máxima en y (rect.yMaximum).

Estos son solo unos pocos comandos, Python tiene una serie de librerías disponibles que permiten realizar muchas consultas y de una complejidad más alta.

## IV. CONCLUSIONES

El uso de la interfaz Python dentro de Qgis nos dio una nueva visión para lograr hacer consultas rápidamente usando código de programación, de esta forma integramos las áreas de SIG (Sistemas de Información Georreferenciada) con la programación .

En los scripts básicos que usamos de ejemplo se muestra la ventaja de poder automatizar algunos procesos de consulta y extracción de información, estos scripts quedan almacenados en el editor de texto y se pueden reutilizar cuantas veces se necesite, disminuyendo el tiempo de las consultas.

Las rutinas almacenadas son fácilmente ajustadas a nuevos proyectos y diferentes capas, en estas tareas repetitivas donde se evidencia el potencial de la programación con Python, desde este punto de vista en programación se convierten en algo similar a la programación de objetos, dado que son reutilizables.

Los comandos diseñados en Pyqgis para ser usado en Qgis son portables y funcionan en aplicaciones GIS desde que tengan la interfaz con Python.

## REFERENCIAS

- [1] Welcome to the QGIS Python API documentation project. Aug 15, 2010 – Feb 3, 2024. nyalldawson, y otros.
- [2] Python Software Foundation. “El tutorial de Python”. Disponible en: <https://docs.python.org/es/3/tutorial/>, 2021-2024.
- [3] Python Software Foundation. “Documentación de la API Python de QGIS”. Disponible en: <https://www.qgis.org/es/docs/index.html>.