Paquete de python para la fusión de imágenes satelitales sobre arquitecturas homogéneas (CPU) y heterogéneas (CPU/GPU) (satellite-images-fusion), utilizando las técnicas: Gram Schmidt, Modulación de altas frecuencias, Transformación High Pass Filter y Valor medio simple

Manual de Instalación

Tabla de contenido

1 Descripción

La fusión de imágenes satelitales es un proceso digital que permite reunir en una imagen procesada la riqueza espectral de una imagen multiespectral y la resolución espacial de una imagen pancromática. Para llevar esto a cabo se cuenta con algunos algoritmos definidos previamente como técnica de valor medio, filtro paso alto, Gram Schmidt y modulación de altas frecuencias, entre otras. A continuación, se da una breve descripción de estas técnicas.

Gram Schmidt

La imagen pancromática se combina con el resto de bandas de menor resolución espacial mediante una transformación matemática de los datos originales. La transformación Gram-Schmidt es una técnica común en álgebra lineal y análisis multivariante; en este caso se aplica con objeto de ortogonalizar las bandas de una imagen digital. La ortogonalización de los datos elimina la información redundante que contienen, algo bastante acusado en las imágenes de satélite, especialmente en las bandas del espectro visible. En el caso de que exista una correlación perfecta entre las bandas de entrada, la ortogonalización Gram-Schmidt produciría una banda en la que todos los píxeles tendrían valor 0. En el caso más realista de que las correlaciones entre las bandas sean muy altas, el resultado será una imagen con valores muy bajos.

Modulación de altas frecuencias

Se trata de una variación de los denominados métodos de fusión en el dominio espacial, cuya idea es la de transferir las altas frecuencias de la imagen de alta resolución a la imagen de baja resolución. Como ya es sabido, las altas frecuencias contienen la información relativa a los detalles de una imagen y pueden extraerse mediante operaciones de filtrado o convolución. Básicamente, estos métodos consisten en la suma de las altas frecuencias de la imagen pancromática a cada banda de la imagen, la eficiencia de este filtro se basa en la existencia de una correlación radiométrica elevada entre los componentes de altas frecuencias de ambas imágenes.

Un caso particular de los métodos espaciales es el llamado modulación de las altas frecuencias (HFM, High Frequency Modulation). Consiste en la multiplicación de cada banda de baja resolución (MS)i, por la imagen de alta resolución (Pan), se obtienen mejores resultados si el filtro paso-bajo se diseña de forma que se ajuste a la función de dispersión puntual (PSF, Point Spread Function) relativa entre ambas imágenes. La PSF es una función de ponderación sobre la señal electrónica que se produce a la salida de los detectores y que depende de factores ópticos, del movimiento de la imagen en su adquisición, del propio detector y de los componentes electrónicos que entran en juego durante el proceso.

Transformación High Pass Filter - HPF

Este método consiste en añadir la información espacial de la banda pancromática a la información multiespectral de menor resolución espacial aplicando un filtro de paso alto en combinación con una operación de álgebra de mapas. El

funcionamiento del algoritmo, descrito en Leica-Geosystems (2006), consta de cinco pasos:

1. Calcular el parámetro R a partir del tamaño del píxel de la capa pancromática (tpan) y de la multiespectral.

2. Aplicar un filtro de paso alto a la imagen pancromática; el tamaño de la ventana de filtrado es proporcional a R. Todos los elementos de la ventana de filtrado toman el valor -1, con la excepción del valor central. Existen tres posibilidades para este valor; el menor de los tres es el que se utiliza por defecto y es el que se ha empleado en este trabajo.

3. Remuestrear la imagen multiespectral al tamaño del píxel de la resolución espacial de la imagen filtrada.

4. Sumar la imagen filtrada a las capas multiespectrales. Pero antes, la imagen filtrada se pondera en función de la desviación típica de la imagen multiespectral y el valor de R, a este factor de ponderación se le denomina W, donde σ_{mis} es la desviación típica de cada una de las bandas; σ_{pan} es la desviación típica de la imagen filtrada y M es un factor que determina la intensidad en la aplicación del filtro. Donde Pout es el píxel de salida de cada una de las bandas multiespectrales ya fusionadas; Pin es el píxel de entrada de cada una de las bandas multiespectrales originales y PHPF es el píxel de la imagen filtrada.

5. Expandir linealmente los niveles digitales (ND) de la imagen multiespectral fusionada; esta operación reescala la imagen resultante, de forma que la media y la desviación típica coincida con las de la imagen original.

Valor medio simple

El método de transformación de valor medio, aplica el promedio de valor medio simple, a cada una de las combinaciones de banda de salida. Por ejemplo, la nueva banda fusionada será la suma de la banda original con la banda pancromática dividido en dos. Así sucesivamente según el número de bandas que conforme la imagen multiespectral.

Por otra parte, una vez se ha aplicado alguno de los algoritmos de fusión, es necesario realizar una evaluación de la riqueza espectral y espacial de la imagen generada por los métodos de fusión expuestos anteriormente, es posible se utilizar algunos índices como mean square error (mse), root mean square error (rmse), Bias y el índice de correlación.

En este orden de ideas, al implementar los algoritmos de fusión de imágenes satelitales en forma serial, es decir, realizando su ejecución exclusivamente en CPU, se presentan tiempos elevados al utilizar imágenes de dimensiones superiores, es por esto que este proyecto busca realizar la implementación de las transformadas mencionadas anteriormente mediante procesamiento heterogéneo CPU/GPU con el fin de optimizar los tiempos de ejecución para este algoritmo. Así mismo, se tiene como objetivo proporcionar herramientas para la comparación en términos de tiempos de ejecución y evaluación de la calidad de la imagen obtenida.

2 Requisitos

Para el funcionamiento de este software, es necesario el siguiente hardware:

1. Tarjeta gráfica NVIDIA, solo si desea realizar la fusión en GPU

Librerías necesarias para ejecutar:

1. Python 3.7 o mayor: https://www.python.org/download/releases/3.6/
2. Numpy 1.21.6: https://pypi.org/project/numpy/
3. Scikit-image 0.18.3: https://pypi.org/project/scikit-image/
4. Scipy 1.7.3: https://pypi.org/project/scipy/
5. Pycuda 2022.1: https://pypi.org/project/pycuda/
6. Scikit-cuda 0.5.3: https://scikit-cuda.readthedocs.io/en/latest/install.html
7. Cupy-cuda111 10.2.0: https://pypi.org/project/cupy/

3 Instalación

Para llevar a cabo la instalación, tenga en cuenta que es necesario que tener instalado Python versión > 3.7, de lo contrario, no será posible llevar a cabo los siguientes pasos. Cuando se instala Python, tendrá instalado PyPI (Python Package Index) por defecto, que es el que permite instalar paquetes de Python, con el comando *pip*.

3.1 Instalar la tarjeta gráfica

Esta sección, solo es necesaria realizarla, si usted desea realizar la fusión de imágenes satelitales mediante la GPU. Primero se debe verificar si el sistema operativo ya reconoce la tarjeta gráfica, para eso se digita el siguiente comando y se verifica que la tarjeta gráfica NVIDIA se encuentre en la lista:

```
lisci | grep -i nvidia
```

En caso de que no aparezca se digita el siguiente comando para actualizar los controladores conectados al computador

```
sudo update-pciids
```

Comprobando que la tarjeta gráfica ya se encuentra instalada se procede a realizar la instalación de los drivers por medio de los siguientes comandos:

```
sudo apt-add-repository ppa:xorg-edgers/ppa

sudo apt-add-repository ppa:ubuntu-x-swat/x-updates

sudo apt-get update

sudo apt-get install nvidia-current nvidia-settings
```

3.2 Instalar CUDA

Para correr la aplicación es necesario instalar CUDA y PyCUDA, el primero se instala por medio de los siguientes dos comandos:

```
sudo apt-get install cuda

sudo apt-get install nvidia-cuda-toolkit
```

Con esto instalado se procede a realizar la configuración de dos variables de entorno, la primera PATH y la segunda CUDA_ROOT de la siguiente forma:

```
export PATH=/usr/local/cuda/bin:$PATH

export CUDA_ROOT=/usr/local/cuda
```

3.3 Instalar dependencias de Python

Para llevar utilizar esta librería en una arquitectura serial haciendo uso de la CPU, se hace necesario mediante el comando *pip* instalar los siguientes módulos de python, de la siguiente forma:

```
pip install numpy==1.21.6

pip install scikitimage==0.18.3

pip install pycuda==2022.1

pip install scipy==1.7.3

pip install scikit-cuda==0.5.3

pip install cupy-cuda111==10.2.0
```

Nota: La version de cupy-cudaXXX dependerá de la versión de CUDA instalada en el sistema operativo.

3.4 Instalación del paquete satellite-images-fusion

Esta librería se encuentra disponible en Python Package Index (PyPI) el cual es un repositorio de software para el lenguaje de programación Python. Por lo tanto, para instalar la librería se debe utilizar el siguiente comando:

pip install *satellite-images-fusion*

Además de esto, se puede encontrar el paquete en el servidor de PYPI en la siguiente dirección.

https://pypi.org/project/satellite-images-fusion/

3.5 Ejemplo

Una vez instalada la librería Sallfus y sus distintas dependencias, se debe abrir la terminal de Python y ejecutar el siguiente código, con el propósito de comprobar su correcto funcionamiento.

```
>>> multi_path = 'multispectral.tif'
>>> pan_path = 'panchromatic.tif'
>>> method = "gram_schmidt"
>>> architecture = "gpu"
>>> geographical_info = True
>>> fusioned_path = f"fusion_{method}_{architecture}.tif"
>>> invoker.generate_fusion_images(multi_path, pan_path, method,
fusioned_path, architecture, geographical_info)
```