

# Transformada PCA sobre computación Heterogénea CPU/GPU

## Manual de Instalación

### Tabla de contenido

1	Descripción.....	2
2	Requisitos.....	2
3	Instalación .....	2
3.1	Instalar la tarjeta gráfica.....	3
3.2	Instalar CUDA y PyCUDA.....	3
3.3	Clonar repositorio .....	4
3.4	Instalar librerías de Python .....	4
3.5	Iniciar aplicación .....	4

# 1 Descripción

La fusión permite la combinación y utilización de datos procedentes de fuentes diferentes, buscando obtener información de “mayor calidad” que la original, la cual dependerá de la aplicación. La fusión de imágenes es una respuesta a la frecuente necesidad de tener en una sola imagen datos de alta resolución espectral y espacial a partir de imágenes de diferente resolución espacial y/o diferentes sensores remotos. La fusión permite obtener información detallada sobre el medio ambiente urbano y rural, útil para una aplicación específica en diferentes estudios geográficos. Las técnicas de fusión de imágenes, como una solución alternativa, se pueden utilizar para integrar el detalle geométrico de una imagen, para este caso de estudio una imagen multiespectral (MULTI) de baja resolución y una pancromática (PAN) de alta resolución, donde se busca producir una nueva imagen de alta resolución espacial y espectral (N-MULT).

Para la realización de la transformada PCA se deben separar las bandas RGB de la imagen MULTI, luego de esto se forma un arreglo al que se le realiza la varianza y covarianza. A continuación, se calculan los componentes principales. En la siguiente etapa se combinan los componentes resultantes para generar una nueva imagen N-MULT.

La transformada PCA se utiliza para obtener una representación simple y con menor dimensión para un conjunto de  $p$  variables correlacionadas. Las componentes principales se obtienen transformando las variables originales a un nuevo conjunto de variables no correlacionadas usando una rotación ortogonal en el espacio  $p$ -dimensional, las primeras componentes resumirán en orden decreciente la mayor cantidad posible de variabilidad de las variables originales.

## 2 Requisitos

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

1. Tarjeta gráfica NVIDIA

También es necesario el siguiente software:

1. Python 2.7: <https://www.python.org/download/releases/2.7/>
2. Numpy 1.14.5: <https://pypi.org/project/numpy/>
3. Pycuda 2017.1.1: <https://pypi.org/project/pycuda/>
4. Skcuda 0.5.2: <https://scikit-cuda.readthedocs.io/en/latest/install.html>
5. Opencv 4.0.0: <https://pypi.org/project/opencv-python/>
6. Scikit-image 0.14.2: <https://pypi.org/project/scikit-image/>
7. Linalg 1.0.4: <https://pypi.org/project/linalg/>

## 3 Instalación

Para llevar a cabo la instalación, tenga en cuenta que es necesario que tener instalado Python versión 2.7, 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

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 y PyCUDA

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
```

Con estas dos variables de entorno se podrá ejecutar el siguiente comando para instalar PyCUDA:

```
pip install pycuda
```

### 3.3 Clonar repositorio

Como primer paso, clone el repositorio de GitHub en su equipo.

```
git clone https://github.com/Parall-UD/ParallelPCA-Transform.git
```

### 3.4 Instalar librerías de Python

Con el comando *pip* debe instalar las librerías de python, de la siguiente forma:

```
pip install numpy
```

```
pip install opencv-python
```

```
pip install pycuda
```

```
pip install scikit-image
```

```
pip install scikit-cuda
```

```
pip install linalg
```

### 3.5 Iniciar aplicación

Para iniciar la aplicación, ubique su carpeta a través de la consola. Ahora, utilice el siguiente comando para realizar la ejecución de la aplicación en CPU:

```
python pca_cpu.py 'parametro1' 'parametro2' salida
```

Y el siguiente comando para realizar la ejecución de la aplicación en GPU:

```
python pca_gpu.py 'parametro1' 'parametro2' salida
```

Como se puede evidenciar tienen la misma cantidad de parámetros, a continuación se hace la descripción de cada uno de ellos

- parametro1: String con la ruta absoluta de la imagen multiespectral a utilizar de tamaño  $A \times A$  donde  $A$  es un número diádico, es decir,  $2^n$ .
- parametro2: String con la ruta absoluta de la imagen pancromática a utilizar de igual tamaño que la multiespectral.
- salida: Nombre de la imagen de salida después de realizar la ejecución de la aplicación.