

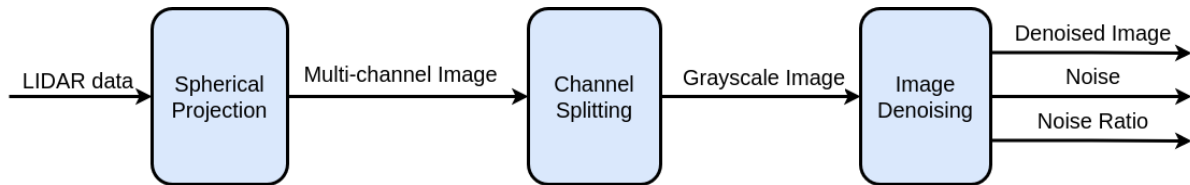
PointCloud Denoising

Denoise LIDAR PointCloud using Spherical Projection and Image Processing.

Updated: 15/06/2021 by Vu-Hoi HUYNH

1. Case study

- Test white noise from LIDAR data, which contains following multidimensional variables:
 - Azimuth angle
 - Elevation angle
 - Distance
 - Intensity
- The idea is to project LIDAR PointCloud from spherical space into a 2D image, then to denoise each channel of the image by Computer Vision methods:



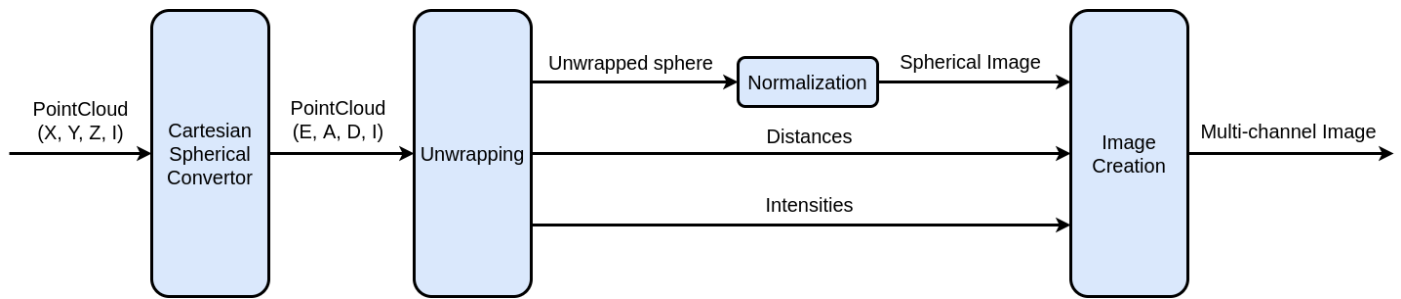
2. LIDAR PointCloud Data Structure

- Azimuth vector: $\mathbf{A} = [a_0 \ a_1 \ \dots \ a_n]^T = [a_j]_{0 \leq j \leq n}^T$
- Distance vector: $\mathbf{D} = [d_j]_{0 \leq j \leq n}^T$
 - Where $d_j = [32a_j + k]_{e_{max} \geq k \geq e_{min}}^T$ and k lies from maximum elevation angle to minimum one.
- Intensity vector: $\mathbf{I} = [i_j]_{0 \leq j \leq n}^T$
 - Where $i_j = [32a_j + k]_{e_{max} \geq k \geq e_{min}}^T$ and k lies from maximum elevation angle to minimum one.

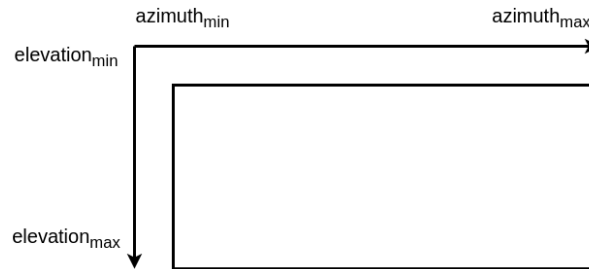
3. Theoretical summary

3.1 Spherical projection

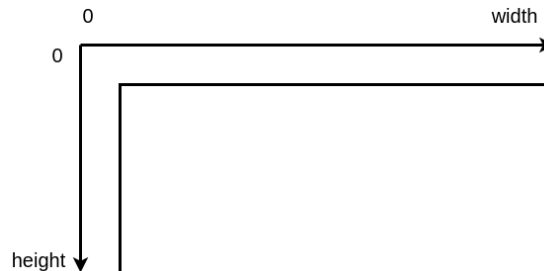
- The principle of spherical projection :



- Where:
 - Unwrapped sphere is a degree-based image with two dimensions of elevation and azimuth angles:



- Spherical image is a pixel-based image with two normalized dimensions of elevation and azimuth angles:



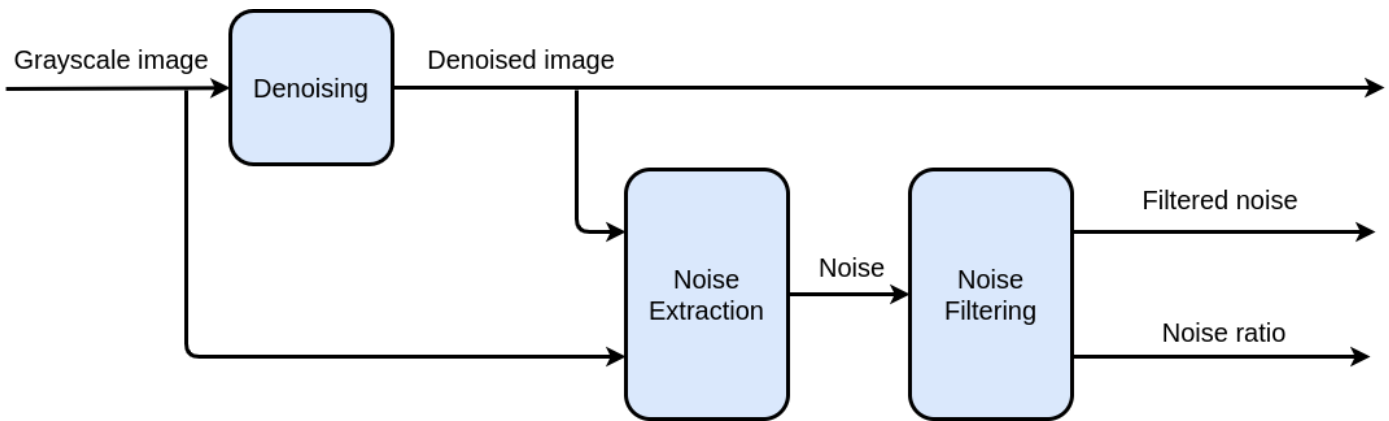
- Multi-channel image contains 4 channels : Elevation, Azimuth, Distance, Intensity
- In our case, PointCloud is already in spherical space so we ignore the first step of Cartesian Spherical Converter .

3.2 Channel Splitting

- The channel splitting is to split multi-channel image into grayscale ones.
- In our case, the Distance and Intensity are studied.

3.3 Image Denoising

- The principle of Image Denoising :



- Where:
 - Denoising uses [fastNlMeansDenoising](#).
 - Noise Filtering uses [Thresh to zero](#) to consider pixels which are smaller than a level, as non-noise pixels and then counters the quantity ratio of $\text{Number of nonzero pixels} / \text{Number of all pixels}$ as a variable which shows the noisy level of the Grayscale image.

4. Usage

- Prerequisites: OpenCV and TinyXML2
- The parameters should be modified inside `parameters/parameterFile.xml`
- Installation and testing:

```
$ mkdir build
$ cd build
$ cmake .. -DCMAKE_INSTALL_PREFIX=../install
$ make
$ ./PointCloud_Denoising ../parameters/parameterFile.xml
```

5. References

- Spherical Projection for Point Clouds
- Computer Vision: Panorama, p.86
- Image Denoising