# **QGIS 3.10 - Geo-analysis Practice**

Lorenzo Amici

Oct 21, 2020

# Contents:

1	Gene	ral Info		3
	1.1	Purpose	e of this documentation	3
	1.2	Target a	audience	3
	1.3	License		3
2	Pren	aration		5
	2.1		QGIS	5
	2.2		bad the data	6
	2.3		a new project	7
	2.5	2.3.1	Import vector data	7
		2.3.2	Import raster data	9
		2.3.3	Layers panel	9
	2.4			10
	2.1	2.4.1	1 5	10
		2.4.2		11
	2.5			13
	2.5	2.5.1		13
		2.5.2	6	17
		2.5.3		17
		2.0.0		1,
3	Vecto	or opera		21
	3.1	Buffer	operations	21
		3.1.1	Single buffer	21
		3.1.2	Multi-ring buffer	23
	3.2	Merge	vector layers	25
	3.3	Overlay	y operations	27
		3.3.1	Union	27
		3.3.2	Intersection	27
		3.3.3	Dissolve	29
		3.3.4	Difference	32
		3.3.5	Symmetrical difference	35
		3.3.6	Spatial join	38
	3.4	Proxim	ity analysis	40
		3.4.1		40
		3.4.2		44
		3.4.3	Distance from point to point	45
		3.4.4	Distance from point to layer	47

	3.5	Network analysis	50
		3.5.1 Shortest path	
		3.5.2 Generate service area	52
	3.6	Kernel density	54
	3.7	Inverse Distance Weighting	57
4	<b>Rasto</b> 4.1	er operations Merge raster	<b>65</b> 65
5	Index	x	71
In	dex		73

This educational material has been developed by Eng. Lorenzo Amici, Dr. Daniele Oxoli, Prof. Maria A. Brovelli and Prof. Fabio Salice of Politecnico di Milano (Italy) with the mentorship and support of Dr. HaeKyong Kang of the Korea Research Institute for Human Settlements, within the project of collaboration between the OSGEO foundation and the UN Open GIS initiative.

Download PDF version

# CHAPTER 1

## General Info

The OSGeo UN Committee promotes the development and use of open source software that meets UN needs and supports the aims of the UN. Following a meeting between OSGeo Board of Directors and the UN GIS team at FOSS4G in Seoul, Korea in September 2015, the Committee has mainly worked on the UN Open GIS Initiative, a project "...to identify and develop an Open Source GIS bundle that meets the requirements of UN operations, taking full advantage of the expertise of mission partners including partner nations, technology contributing countries, international organisations, academia, NGOs, private sector. The strategic approach shall be developed with best and shared principles, standards and ownership in a prioritized manner that addresses capability gaps and needs without duplicating efforts of other Member States or entities. The UN Open GIS Initiative strategy shall collaboratively and cooperatively develop, validate, assess, migrate and implement sound technical capabilities with all the appropriate documentation and training that in the end provides a united effort to improve the effectiveness and efficiency of utilizing Open Source GIS around the world."

# 1.1 Purpose of this documentation

This educational material is designed as a step-by-step software learning guide for QGIS. It contains the description of some functions for environmental analysis of vector and raster data, and a hands-on example for each function described. The purpose of this quick start document is to introduce the user in the use of the main algorithms contained in QGIS and used in environmental analysis.

# 1.2 Target audience

The primary target audience for this documentations is intermediate QGIS users with a background of data analysis and management in a GIS environment

## 1.3 License

This educational material was written by Eng. Lorenzo Amici and Dr. Daniele Oxoli of Politecnico di Milano.

It is distributed according to the CREATIVE COMMONS deed: Attribution - NoDerivs 2.0. According to this license type you are free to:

- copy, distribute, display and perform the work
- make commercial use of the work

Under the following conditions:

- you must attribute the work in the manner specified by the author or licensor
- you may not alter, transform, or build upon this work.

For any reuse or distribution, you must make clear to others the license terms of this work. Any of these conditions can be waived if you get permission from the copyright holder. Your fair use and other rights are in no way affected by the above. This is a human-readable summary of the Legal Code (the full license) that can be consulted at this website.

# CHAPTER 2

# Preparation

In this section we will show all the necessary setup steps in order to follow along the QGIS exercise.

# 2.1 Install QGIS

Download QGIS 3.10 (Long Term Release) for Windows 64bit at https://qgis.org/downloads/ QGIS-OSGeo4W-3.10.7-1-Setup-x86\_64.exe. Other versions are available in the QGIS Download page.

**Note:** After downloading QGIS, you will have various option to launch the program, as you see in this image:



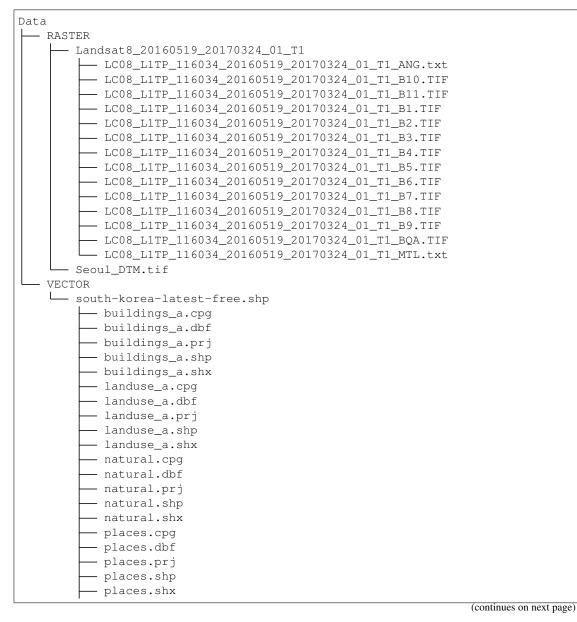
In order to complete this exercise be sure to launch the one with GRASS since we will need some functionalities that are only available using the tools that GRASS offers.

## 2.2 Download the data

The dataset used in the following exercises can be downloaded here; it covers an area around the city of Seoul in South Korea and contains three different type of information:

- Landsat8 images in the folder LC08\_L1TP\_115034\_20180721\_20180731\_01\_T1: it contains all the 11 available bands of the Landsat8 images in *WGS84/UTM zone 52N* coordinate system *EPSG:32652*.
- Aster Digital Elevation map (DTM at a resolution of 20 m) Seoul\_DTM.tif: a geotif in LongLat WGS84 coordinate system *EPSG:4326*.
- OpenStreetMap vector dataset in the folder south-korea-latest-free.shp: these are 8 shapefiles downloaded from the OSM website. The files are in LongLat WGS84 coordinate system EPSG:4326.

Here we illustrate the Data folder tree structure:



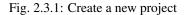
(continued from previous page)

pofw_a.cpg
pofw_a.dbf
— pofw_a.prj
pofw_a.shp
— pofw_a.shx
— pois_a.cpg
— pois_a.dbf
— pois_a.prj
— pois_a.shp
— pois_a.shx
README
— roads.cpg
- roads.dbf
— roads.prj
- roads.shp
- roads.shx
water_a.cpg
water_a.dbf
water_a.prj
water_a.shp
└── water_a.shx

## 2.3 Create a new project

Before starting with the analysis of the data, we have to set up a new Project. To do so, run QGIS with GRASS and click on *New Project* (or press Ctrl+N).

Q Untitled Project - QGIS			-	٥	×
Project Edit View Layer Settings Plu	ugins Ve	ttor Baster Database Web SCP Processing Help			
New Ctrl+	N	- 回 - 回 - 回 - 回 - 回 - U - 回 - U - U - U			
New from Template	•	· · ◎ ① ~ ② ② ~ · · · · · · · · · · · · · · · ·			
Den Ctrl+	0				
Open From	•	<~			
Open <u>R</u> ecent	•				



#### 2.3.1 Import vector data

To import vector data, go to *Layer->Add layer->Add vector layer* or use the Browser panel (usually placed above the Layer Panel on the left side of the screen; if not, you can enable it by clicking *View->Panels* and tick "Browser panel"). In the Browser, you can search the data folder and simply drag and drop the files in the map.

For this excercise, we will use the following layers:

- buildings\_a.shp
- landuse\_a.shp
- natural.shp
- places.shp
- pofw\_a.shp
- pois\_a.shp

Q Untitled Project - Q										-	٥	×
Project Edit View	Layer Settings Plugins Vector Raste	er <u>D</u> atabase <u>W</u> ei Ctrl+L										
🗋 🗁 🖶 🛃 [	Data Source Manager	Ctri+L	A 🖪 💾 🗖 🏹 I 🔍 Q → 🖪			- <u>m</u> -						
🛛 🦛 🗞 🔏	Create Layer Add Laver	•	V, Add Vector Layer	Ctrl+Shift+V	👌 🔍 🛠 🛛	?						
👌 💘 📜 12	Embed Layers and Groups		Add Raster Laver	Ctrl+Shift+R								
	Add from Layer Definition File		Add Delimited Text Layer									
6-9-4-	Copy Style		R Add PostGIS Layers	Ctrl+Shift+D								ØX
Browser	Paste Style		🖉 Add SpatiaLite Layer	Ctrl+Shift+L					Processing Toolbox 🎭 🔩 🕓 📄 🔍 🔧			ØX
Favorites	Copy Laver		Add MSSQL Spatial Layer	Ctrl+Shift+M					9. Search			
► A Home	Paste Layer/Group		Add DB2 Spatial Layer	Ctrl+Shift+2					Recently used			
C\	Open Attribute Table	F6	Add Oracle Spatial Layer	Ctrl+Shift+O					<ul> <li>Q Cartography</li> </ul>			
🍄 GeoPackage	/ Toggle Editing	10	Add/Edit Virtual Layer						<ul> <li>Q Database</li> </ul>			
<ul> <li>SpatiaLite</li> <li>PostGIS</li> </ul>	Save Laver Edits		R Add WMS/WMTS Layer	Ctrl+Shift+W					Q File tools     Q Graphics			
MSSQL	// Current Edits	•	R Add ArcGIS MapServer Layer						Q Interpolation			
Oracle	Save As		Reference Add WCS Layer						<ul> <li>Q Layer tools</li> </ul>			
DB2	Save As Layer Definition File		😪 Add WFS Layer						Q Network analysis     Q Raster analysis			
	Remove Layer/Group	Ctrl+D	Add ArcGIS FeatureServer Layer						Q Raster analysis     Q Raster terrain analysis			
🕨 🌐 WCS	Duplicate Layer(s)	Cureb							Q Raster tools			
W WFS	Set Scale Visibility of Laver(s)								<ul> <li>Q Vector analysis</li> </ul>			
<ul> <li>OWS</li> <li>ArcGisMapServ</li> </ul>		Ctrl+Shift+C							Q Vector creation     Q Vector general			
ArcGisFeatureS	Set Project CRS from Layer	curromitte							Q Vector geometry			
SeoNode	Layer Properties								<ul> <li>Q Vector overlay</li> </ul>			
Layers	Filter	Ctrl+F							<ul> <li>Q Vector selection</li> </ul>			
🗸 🕼 🕸 👗 🖓 – 🖬	Labeling	Curri							Q Vector table     DataPlotly			
	Show in Overview								<ul> <li>Ø dzetsaka</li> </ul>			
	Show All in Overview								🕨 🚋 GDAL			
	Hide All from Overview								<ul> <li>▶</li></ul>			
			-						<ul> <li>♥ Visibility analysis</li> <li>♥ Visualist</li> </ul>			
Q Type to locate (Ctrl+K	)				Coordinate 125.291	,38.944 🔏 🤋	Scale 1:1334153 ¥	Magnifier :	100% 🗘 Rotation 0,0 °	🗘 🗸 Render 🐗	EPSG:4326	Q

Fig. 2.3.1.1: Add a vector layer

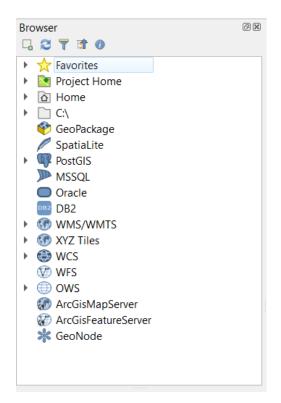


Fig. 2.3.1.2: The Browser panel

- roads.shp
- water\_a.shp

#### 2.3.2 Import raster data

To import raster data, you can go to *Layer->Add layer->Add raster layer* or drag and drop them from the Browser panel. For this exercise, you can add the following data:

• Seoul\_DTM.tif

#### 2.3.3 Layers panel

The Layers panel is a useful way to keep track of all the layers currently loaded in our Project. It's usually placed below the Browser panel, but if you don't have it activated you can do so by clicking on *View->Panels* and tick "Layers".

Once you add all the data it should look like this:

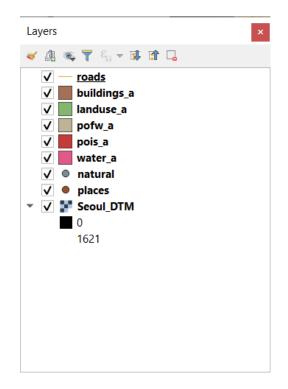
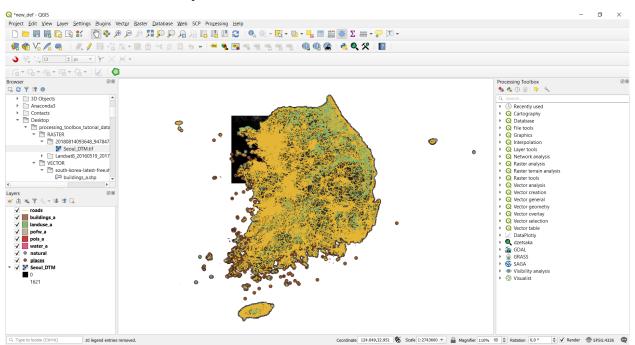


Fig. 2.3.3.1: The Layers panel

The Layers panel can also be used to choose which layers we want to see in our map: just by unticking a layer we hide it from our map. Also, we can choose the order of the layers, because layers can overlap and hide portions of each other, so it's important to decide which one is visualized on top of the others. To do so, you can select the layer you want to change and use the arrow symbols, or simply drag the layer in the position you want it to be.

**Note:** If the Project CRS (that can be seen in the bottom right corner of the window) is not specified, when importing the first layer QGIS will set the Project CRS to the CRS of that first layer. From then onwards, QGIS automatically

reprojects any imported layer in the projection of the Project but only for visualization, the data remain in the original projection.



Once all the data are added, the map should look like this:

We can now save our project (*Project->Save* or Ctrl+s) so that you will have all the added layers and progress always available by just opening the saved project. Remember to do this from time to time during the exercise.

**Note:** When saving a project, QGIS creates a . qgz file, that represents the saved project. Note that this file does not contain directly all the added layers, but it records the path necessary to reach each one of the layers, enabling QGIS to include them when opening the saved project.

## 2.4 Manipulate CRS of the project and the data

The data we use in our geo-analyses often come from different sources, and therefore also have a different Coordinate Reference System (CRS). In order to homogenize the works and assure that all the tools work correctly, it is recommended to reproject all the data in the same CRS.

### 2.4.1 Reprojecting vector layers

This tool is available at *Vector->Data management tools->Reproject layer*. It provides a function that reprojects a vector layer, creating a new layer with the same features as the input one, but with geometries reprojected to a new CRS. The required input parameters are:

- *Input layer*: the vector layer to be reprojected (in the example :file'buildings\_a')
- *Target CRS*: the target Coordinate Reference System. You can choose from the recent ones in the dropdown menu or click the icon to choose between all the available ones. Our choice will be *EPSG*:32652 *WGS* 84 / *UTM zone* 52N

• *Reprojected*: the path and the name of the output raster layer. Note that if left empty a temporary layer will be created

	4	
Parameters Log	Reproje	ect layer
Input layer	This algorithm	n reprojects a
Selected features only	features as th	v layer with the ne input one, b eprojected to a
Target CRS	Attributes are algorithm.	e not modified
EPSG:32652 - WGS 84 / UTM zone 52N	▼	
Reprojected		
[Create temporary layer]	Create Tempo	
✓ Open output file after running algorithm	Save to File	
	Save to GeoP	ackage
	Save to PostG	JIS Table
	Change File E	Incoding (Sy
0%		
Run as Batch Process	Run	Close

Fig. 2.4.1.1: Reproject layer function window

#### 2.4.2 Reprojecting raster layers

Available at *Raster->Projections->Warp (reproject)*, it provides a function that reprojects a raster layer. The tool requires as input:

- Input layer: the raster layer to be reprojected (in the example the Seoul\_DTM)
- *Target CRS*: the target Coordinate Reference System. You can choose from the recent ones in the dropdown menu or click the icon to choose between all the available ones. Our choice will be *EPSG*:32652 *WGS* 84 / *UTM zone* 52N
- *Resampling method to use*: the method to be used for resampling the data, we will use the *Nearest Neighbor*

• *Reprojected*: the path and the name of the output raster layer. Note that if left empty a temporary layer will be created

🔇 Warp (Reproject)	
Parameters Log	
Input layer	
Seoul_DTM [EPSG:4326]	
Source CRS [optional]	
Target CRS	
EPSG:32652 - WGS 84 / UTM zone 52N	
Resampling method to use	
Nearest Neighbour	
Nodata value for output bands [optional]	
Not set	
Output file resolution in target georeferenced units [optional]	
Not set	
Advanced parameters	
Reprojected	
[Save to temporary file]	
✓ Open output file after running algorithm	Save to a Temporary F
GDAL/OGR console call	Save to File
gdalwarp -t_srs EPSG:32652 -r near -of GTiff C:/Users/lawfr/Desktop/processing_toolbox_tutorial_dataset/RASTER/2018081	Change File Encoding
Users/lawfr/AppData/Local/Temp/processing_3be55fc07f624efa81224bca407f2aab/bf55cf6d10a9487c9faee3101e698a45/OUT	P01.ur
0%	
Run as Batch Process	Run Close

Fig. 2.4.2.1: Reproject raster function window

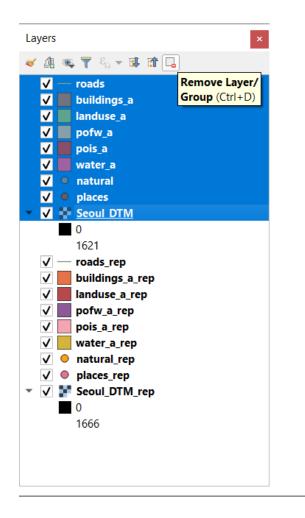
In order to continue with the tutorial please reproject all the imported vector and raster layers following the examples above. Once all the layers are reprojected we need to check if the CRS of the project is the same as the data: to do so, click on the CRS setting button in the bottom right of the window and select also here *EPSG:32652 - WGS 84 / UTM zone 52N*.

**Note:** In this exercise we will refer to the reprojected layers as *originalname\_rep*. You can create your new layers with the same name in order to be consistent with the naming.

**Note:** After reprojecting all the layers be sure to delete the original layers. You can do this by simply selecting the layers you want to delete in the Layer panel and then click on the Remove layer/Group button.

C. Type to locate (244.886,38.169 🛞 Scale 1:1528128 💌 🚔 Magnifier 100% 🗘 Rotation (0.0 ° 👂 🖉 Reter 💮 ETSC:4325 🔍

Fig. 2.4.2.2: You can see the Project CRS button highlighted in red



## 2.5 Clip all the data to the study area

Often the data provided is in a wider area than the one needed, so we can define an area of interest and clip all the layers to that same area; in this way we shorten computational times and also provide consistency to the data.

#### 2.5.1 Create the working area

We will create a vector polygon layer, in the shape of a rectangle, that will represent the area we are considering in this tutorial. In order to do so:

- Create a vector layer using Layer->Create Layer->New shapefile layer or use the shortcut symbol
  - A 🕄
- Specify its characteristics as follows:
  - File name: path/area\_of\_interest.shp

- File encoding: System
- *Geometry type*: Polygon
- The CRS should be by default set to the one of the maps but check it to be *EPSG: 32652 WGS 84 / UTM zone 52N*

🔇 New	Shapefile Layer								
File nam	e		C:\Users\lawfr\De	sktop\processing_toolbox_tuto	rial_dataset\area_o				
File enco	ding		System						
Geometr	y type		Polygon						
			Include Z dime	Include Z dimension Include M va					
			EPSG:32652 - WG	S 84 / UTM zone 52N					
New Fie	ld								
Name									
Туре	abc Text data								
Length	80	Precision							
		Ad	d to Fields List						
Fields Li	st								
Name	Туре		ength	Precision					
id	Integ		0	riecision					
					ОК				

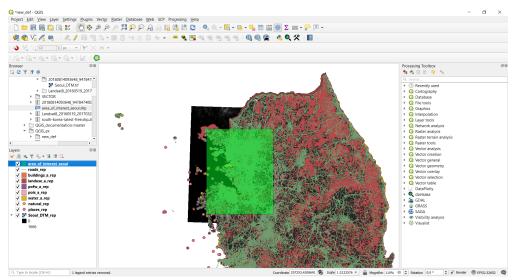
• Click "Ok"

Once you created the layer, you need to add the polygon representing the working area:

- Right click on area\_of\_interest in the Layers panel and select "Toggle editing"
- Right-click anywhere on the top toolbar (or go to *View->Toolbars* and add it from there) and enable the "Shape digitizing toolbar". This way we can add a perfect rectangle polygon to our shapefile
- Click on "Add rectangle from extent"

🔇 *Untitled Project - QGIS					
Project <u>E</u> dit <u>V</u> iew	Layer Settings Plugins Vector Raster Database Web				
🗋 📄 🖥 🛃 🖡	, 👧 🔍 🔍 💐 🖏 🔁 🗨 🛸 🖞 🗄 🎦				
🛛 🐫 🎕 V° 🖉 (	🖏 🗄 🥢 🕞 📆 🎘 🗕 🛍 🕯 🖬 🔞				
👌 💘 📜 12	px ▼ Y × × ▼				
/ <mark>· · Q</mark> · <i>Q</i> ·					
Browser	Add <u>R</u> ectangle from Center and a Point				
🗔 😂 🝸 😭 🕖	📮 Add Rectangle from Extent				
🕨 🛧 Favorites	Gamma Add Rectangle from 3 Points				
Home					

• Draw a rectangle in the area around Seoul by left-clicking to start drawing from an angle and then right-clicking when the size is the desired one (not too big, take the picture below as reference)



• Click on "Save layer edits" and toggle editing off

Now that you have your working area layer we can clip all our layers to it.

## 2.5.2 Clipping vector layers

Available at *Vector->Geoprocessing tools->Clip*. It provides an algorithm that clips a vector layer using the features of another polygon layer. Only the parts of the features in the Input layer that fall within the polygons of the Overlay layer will be added to the resulting layer. The attributes of the features are not modified, although properties such as area or length of the features will be modified by the clipping operation. The input parameters are:

- *Input layer*: the vector layer to be clipped (in the example landuse\_a\_rep). You can also choose to clip only the selected features of the vector layer if there are any
- Overlay layer: the area\_of\_interest layer
- *Clipped*: the path and the name of the output vector layer. Note that if left empty a temporary layer will be created

### 2.5.3 Clipping raster layers

Available at *Raster->Extraction->Clip raster by mask layer*, it provides an algorithm that clips a raster layer using a vector layer as a mask. The input parameters are:

- *Input layer*: the raster layer to be clipped (in the example Seoul\_DTM\_rep)
- *Mask layer*: the area\_of\_interest layer
- *Clipped (extent)*: the path and the name of the output raster layer. Note that if left empty a temporary layer will be created

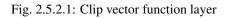
In order to continue with the tutorial please clip all the vector and the DTM raster layer following the examples above.

**Note:** In this exercise we will refer to the clipped layers as *originalname\_clip*. You can create your new layers with the same name in order to be consistent with the naming

Note: After clipping all the layers you can delete the starting layers from the Layers panel.

The final configuration of the application and data should be like the one in the next image.

Parameters Log	<sup>1</sup> Clip
Input layer  Input layer Input	This algorithm clips a vect the features of an addition Only the parts of the features layer that fall within the pu
Overlay layer	Overlay layer will be adde layer.
C area_of_interest_seoul [EPSG:32652]	The attributes of the feature
Selected features only Clipped	modified, although proper or length of the features v by the clipping operation. properties are stored as a attributes will have to be
[Create temporary layer]	Create Temporary Laye
✓ Open output file after running algorithm	Save to File
	Save to GeoPackage
	Save to PostGIS Table
	Change File Encoding (
0%	



🔇 Clip Raster by Mask Layer		
Parameters Log		
Input layer		
Seoul_DTM_rep [EPSG:32652]		
Mask layer		
area_of_interest_seoul [EPSG:32652]		,
Selected features only		
Assign a specified nodata value to output bands [optional]		
Not set		
Create an output alpha band		
$\checkmark$ Match the extent of the clipped raster to the extent of the mask layer		
Keep resolution of output raster		
Advanced parameters		
Clipped (mask)		
C:/Users/lawfr/Desktop/QGIS_ex/new_def/Seoul_DTM_clip.tif		
✓ Open output file after running algorithm		
GDAL/OGR console call		
gdalwarp -of GTiff -cutline C:/Users/lawfr/Desktop/processing_toolbox_tutorial_dataset/area_of_interest_seoul.shp -cl area_ lawfr/Desktop/QGIS_ex/new_def/Seoul_DTM_rep.tif C:/Users/lawfr/Desktop/QGIS_ex/new_def/Seoul_DTM_clip.tif	of_interest_seoul	-crop_to_cutli
0%		
Run as Batch Process	Run	Close

Fig. 2.5.3.1: Clip raster by mask layer function window

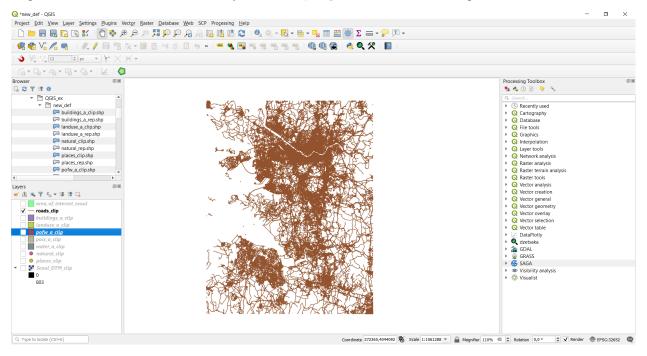
roject Edit View Layer Settings Plugins Vector Baster	jatabase Web SCP Processing Help 💯 💯 🗛 🔏 🔚 🖪 🖄 🖉 🔍 🔍 = 👯 = 🤮 = 🎭 🛅 🗟 🎆 Σ 🚍 = 🖓 🗊 =	
	· · · · · · · · · · · · · · · · · · ·	
3 ₩ 1,12 ₽ px - Y × × -		
fa + Qa + Qa + Qa + 🛃 🧔		
owser @ 🛛		Processing Toolbox
27 1 0		- 🎭 🔩 🕓 🖹 I 🦻 I 😽 I
V CGIS_ex		Q. Search
✓ T new_def		
buildings_a_clip.shp		Q Cartography
🟳 buildings_a_rep.shp		Q Database
Ianduse_a_clip.shp		Received by File tools
Ianduse_a_rep.shp		Q Graphics
🏳 natural_clip.shp		Q Interpolation
🟳 natural_rep.shp		Q Layer tools
places_clip.shp		Retwork analysis
Places_rep.shp		<ul> <li>Q Raster analysis</li> </ul>
🗭 pofw_a_clip.shp 🚽		<ul> <li>Q Raster terrain analysis</li> </ul>
~~ >		Raster tools
vers @ 8		<ul> <li>Q Vector analysis</li> </ul>
· /// ≪ ▼ % + ⊯ m ⊡ □		<ul> <li>Q Vector creation</li> </ul>
area of interest seoul		Vector general
✓ — roads clip		Vector geometry
✓ buildings_a_clip		Q Vector overlay
✓ landuse a clip		Q Vector selection
V pofw_a_clip		Q Vector table
V pois_a_clip		DataPlotly
V water a clip		• Q dzetsaka
✓ ● natural_clip		Amage GDAL     Age GRASS
✓ ● places clip		GRASS     GRASS     GRASS
V F Seoul_DTM_clip		W Visibility analysis
0		Si Visualist
803		, () visualist
	and the second	

# CHAPTER 3

## Vector operations

## 3.1 Buffer operations

We will now focus on the analysis and manipulation of line shapefiles. To do so, let's consider only the roads\_clip shapefile. You can turn-off all the other layers in the *Layers panel* to have a more compact visualization of the data.

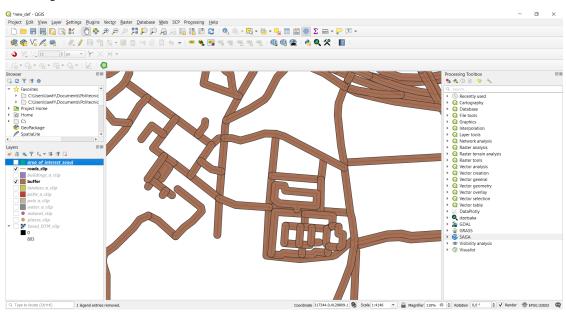


## 3.1.1 Single buffer

A very common operation with line shapefiles is the buffer, which allows to create an area within a specified distance from features. Note that in QGIS a buffer can be done also around point or polygon shapefiles. To create a buffer, click on *Vector->Geoprocessing Tools->Buffer* and specify the input parameters as follows:

- Input layer: the roads\_clip layer
- Distance: the desired distance (in our case 10m). Note that you can also specify the unit of measure
- *Segments*: the number of line segments to be used to approximate a quarter circle when creating rounded offsets (in our case 5)
- *Buffered*: the path and the name of the output vector layer. Note that if left empty a temporary layer will be created

🔇 Buffer		
Parameters Log	4	Buffer
Input layer		Butter This algorithm comp all the features in ar fixed or dynamic dist The segments parar number of line segments approximate a quart creating rounded off The end cap style param round, miter or bever used when offsetting The miter limit parar applicable for miter
Round Miter limit 2,00000	<ul> <li>▼</li> <li>↓</li> </ul>	controls the maximu offset curve to use v mitered join.
Dissolve result Buffered [Create temporary layer]		
Create temporary layer]  ✓ Open output file after running algorithm		
0%		
Run as Batch Process		Run

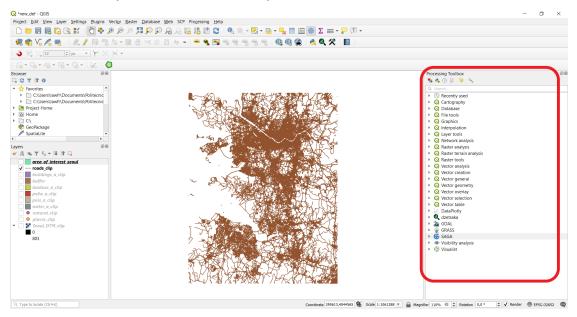


This will create a buffer area around our roads layer; if you zoom on your map, the result should look the following.

### 3.1.2 Multi-ring buffer

You can also create multiple buffers around the same features, within a specified distance.

To select the above function, we introduce here a very powerful instrument in QGIS: the Processing Toolbox. To add it to your window, go to *View->Panels* and tick "Processing Toolbox panel". It should appear on the right side of your screen; this panel contains all the functions of QGIS, and in particular, the search bar is very useful to find the function you need.

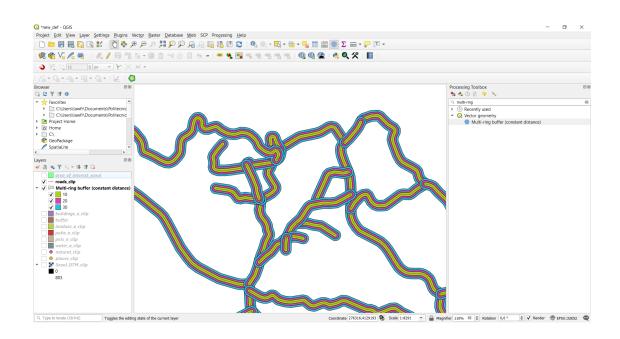


In this case, you can search for "Multi-ring" and select the function *Multi-ring buffer (constant distance)*. The input parameters are:

- *Input layer*: the roads\_clip layer
- Number of rings: in our case 3
- Distance between rings: in our case 10m
- *Multi-ring buffer (constant distance)*: the path and the name of the output vector layer. Note that if left empty a temporary layer will be created

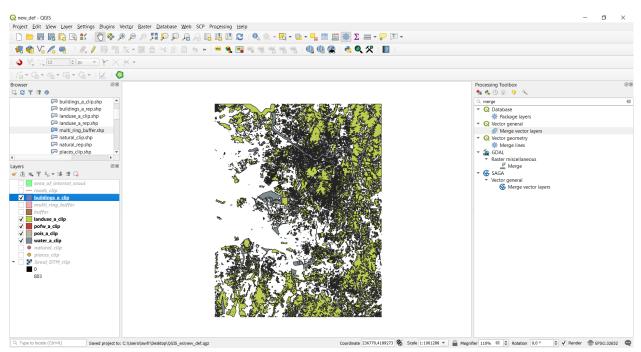
Q Multi-Ring Buffer (Constant Distance)	
Parameters Log	Multi-ring b
Input layer	(constant d
√° roads_clip [EPSG:32652]	This algorithm comp
Selected features only	('donuts') buffer for input layer, using a f
Number of rings	distance and rings n
3	
Distance between rings	
10,000000 🚳 🗘 meters 👻 🗲	
Multi-ring buffer (constant distance)	
C:/Users/lawfr/Desktop/QGIS_ex/new_def/multi_ring_buffer.shp	
✔ Open output file after running algorithm	
0%	
Run as Batch Process	Run Cl

This will create three buffer areas around our roads layer; if you zoom on your map, and if properly styled, the result should look like this:



# 3.2 Merge vector layers

Let's now consider the polygons layers: in particular we will need the buildings\_clip, landuse\_a\_clip, pois\_a\_clip, pofw\_a\_clip and water\_a\_clip layers, so you can hide all the others in the map.



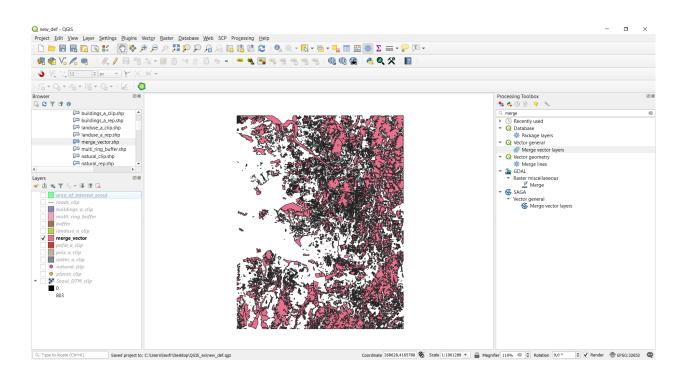
The Merge vector layers function, available at *Processing Toolbox->Vector General->Merge vector layers*, provides an algorithm that combines multiple vector layers of the same geometry type into a single one. If the attributes tables are different, the attribute table of the resulting layer will contain the attributes from all input layers. New attributes will be added for the original layer name and source. We will use it to combine the landuse\_a\_clip and the water\_a\_clip in order to get a general layer describing both land and water features. To do so, the input parameters are:

- Input layers: click on the icon on the left and select landuse\_a\_clip and water\_a\_clip
- Destination CRS: select the Project CRS, that is EPSG:32652 WGS84 / UTM zone 52N
- *Merged*: the path and the name of the output vector layer. Note that if left empty a temporary layer will be created

🔇 Merge Vector Layers	×
Parameters Log	Merge vector layers
Input layers  0 elements selected  Destination CRS [optional]  Project CRS: EPSG:32652 - WGS 84 / UTM zone 52N  Merged  C:/Users/lawfr/Desktop/QGIS_ex/new_def/merge_vector.shp  ✓ Open output file after running algorithm	Image: Solution of the sector is a single one.         Image: Solution of the sector is a single one.         Image: Solution of the sector is a single one.         Image: Solution of the sector is a single one.         Image: Solution of the sector is a single one.         Image: Solution of the sector is a single one.         Image: Solution of the sector is a single one.         Image: Solution of the sector is a single one.         Image: Solution of the sector is a single one.         Image: Solution of the sector is a single one.         Image: Solution of the sector is a single one.         Image: Solution of the sector is a single one.         Image: Solution of the sector is a single one.         Image: Solution of the sector is a single one.         Image: Solution of the sector is a single one.         Image: Solution of the sector is a single one.         Image: Solution of the sector is a single one.         Image: Solution of the sector is a single one.         Image: Solution of the sector is a single of the sector is a single one.         Image: Solution of the sector is a single one.         Image: Solution of the sector is a single one.         Image: Solution of the sector is a single one.         Image: Solution of the sector is a single one.         Image: Solution of the sector is a single one.         Image: Solution of the sector is a single on
0%	Cancel
Run as Batch Process	Run Close Help

Fig. 3.2.1: Merge function window

As you can see this operation creates a new vector that contains features from both the landuse and water bodies layers:



# 3.3 Overlay operations

#### 3.3.1 Union

The Union function, available at *Vector->Geoprocessing->Union*, provides an algorithm that checks overlapping features within an input layer and creates separate features for overlapping and non-overlapping parts. The area of overlap will create as many identical overlapping features as there are features that participate to that overlap.

In this exercise, we will perform an Union between the pois\_a\_clip layer (points of interest) and the buildings\_a\_clip layer, so that we will get a layer representing both residential buildings and interest/public buildings. The input parameters are:

- Input layer: pois\_a\_clip layer
- Overlay layer: buildings\_a\_clip layer
- *Union*: the path and the name of the output vector layer. Note that if left empty a temporary layer will be created

The result, if zoomed, should look like the following:

#### 3.3.2 Intersection

We can now look at the Intersection function. It is available at *Vector->Geoprocessing Tools->Intersection*, and it provides a function that extracts the overlapping portions of the features of two layers and assigns these portions the attribute of both layers. We use it to see which buildings are also religious buildings. The input parameters are:

- *Input layer*: buildings\_a\_clip layer
- Overlay layer: pofw\_a\_clip layer

Q Union		
Parameters Log	•	Union
Input layer		This algorithm checks overla
♥ pois_a_clip [EPSG:32652] ♥ ♥		features within the Input lay separate features for overla
Selected features only		overlapping parts. The area create as many identical over
Overlay layer [optional]		features as there are featur participate in that overlap.
Selected features only		An Overlay layer can also be which case features from ea
Union		split at their overlap with feat other one, creating a layer of
C:/Users/lawfr/Desktop/QGIS_ex/new_def/union.shp		the portions from both Input layers. The attribute table o
✓ Open output file after running algorithm		layer is filled with attribute respective original layer for
		overlapping features, and a from both layers for overlap
		non both dyers for overlap
0%		
Run as Batch Process		Run Close

Fig. 3.3.1.1: Union function window

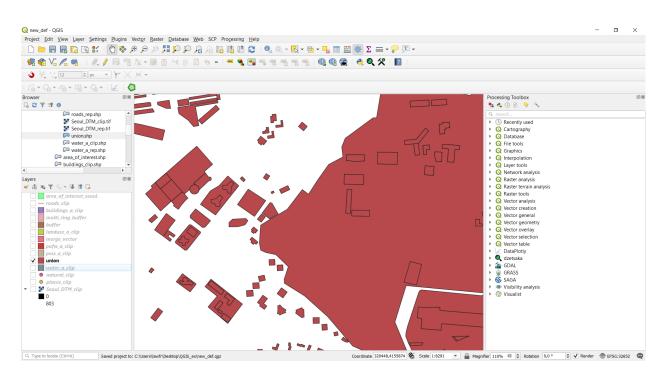


Fig. 3.3.1.2: As you can see now both residential and interest/public buildings are contained in the same vector layer

• *Intersection*: the path and the name of the output vector layer. Note that if left empty a temporary layer will be created

The results should look like the following picture:

#### 3.3.3 Dissolve

We will now use the Dissolve function applied to the landuse\_a\_clip layer. The Dissolve function, available at *Vector->Geoprocessing tools->Dissolve*, provides an algorithm that takes a vector layer and combines their features into new features. One or more attributes can be specified to dissolve features belonging to the same class (having the same value of a specific attribute); if no attribute is selected, all features will be dissolved into a single one. We will create a new landuse layer that has as many features as the different types of land use. The input parameters are:

- Input layer: the landuse\_a\_clip layer
- Dissolve fields: click the icon on the left and select the "fclass" attribute
- *Dissolved*: the path and the name of the output vector layer. Note that if left empty a temporary layer will be created

If you now check the attribute table of the newly created layer it should have only a few features, each corresponding to one type of land use.

Q Intersection		
Parameters Log	• ]	Intersection
Input layer	7	This algorithm extracts the
→ buildings_a_clip [EPSG:32652]		portions of features in the I Overlay layers. Features in
Selected features only	I	intersection layer are assign
Overlay layer		attributes of the overlapping both the Input and Overlay
♥ pofw_a_clip [EPSG:32652]		
Selected features only		
Input fields to keep (leave empty to keep all fields) [optional]		
0 elements selected		
Overlay fields to keep (leave empty to keep all fields) [optional]		
0 elements selected		
Intersection		
C:/Users/lawfr/Desktop/QGIS_ex/new_def/intersection.shp		
✔ Open output file after running algorithm		
0%		
Run as Batch Process		Run Close

Fig. 3.3.2.1: Intersection function window

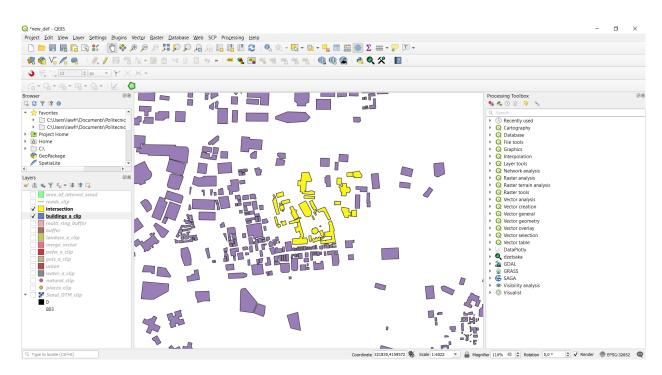


Fig. 3.3.2.2: As you can see, the intersection layer (yellow) only contains the buildings that also are religious buildings

<ul> <li> (2)</li> <li></li></ul>	This algorithm takes a vector layer and combines their features into new feature One or more attributes can be specified dissolve features belonging to the same class (having the same value for the specified attributes), alternatively all features can be dissolved in a single one All output geometries will be converted to multi geometries. In case the input is a
	polygon layer, common boundaries of adjacent polygons being dissolved will g
Q Multiple selection         osm_id         code         ✓ fclass         name	
	Cance
	Multiple selection     osm_id     code     v fclass     name

Fig. 3.3.3.1: Dissolve function window

G	Dissolved :: Featu	ures Total: 19, Filter	ed: 19, Selected: 0	_	×
/	1 🖶 😂 i 🛱 i	i 🗠 i 🛙 i 😜	= 🔊 🔩 🍸 🔳	🏘 🔎 i 🖪 🖩 🔛	Q.
	osm_id	code	fclass	name	
1	26271764	7212	retail		
2	26271759	7203	residential		
3	26628717	7211	recreation_grou	ë¹ì§€ìš′ë <b>Å</b> ™ìž¥	
4	174111566	7207	allotments		
5	196319894	7217	scrub		
6	38316679	7205	farm	ë† <b>ì</b> ìŒì§"í¥¥ì²ì	
7	26259548	7202	park	ì⊄…ë¬~	
8	120027471	7216	vineyard		
9	452702676	7210	nature_reserve	ìž <b>É</b> ì—°í∙™ìŠµìž¥	
10	32007357	7201	forest	서ì~¤ë¦‰	
11	41943278	7213	military	ìn∿ <b>É</b> ê³µí∙-	
12	223871785	7206	cemetery	천ì∙^ê³µì∍Éë¬~	
13	363531897	7214	quarry		
14	472004967	7219	heath	í™″성 ê³ ì •ë¦¬	
15	38141486	7218	grass	ë§Œì" <b>¥</b> ê³µì∍ <b>É</b>	
16	38141719	7204	industrial	KT&G	
17	119423303	7208	meadow		
18	115698838	7215	orchard		
19	31856777	7209	commercial	ë§ <b>¥</b> ì∍Éì⊲œìž¥	
			-		
T	Show All Features				3

### 3.3.4 Difference

We can now use the newly created landuse\_dissolved layer to perform a Difference operation. Difference is a function available at *Vector->Geoprocessing Tools->Difference*, that extracts features of an Input layer that fall outside features in the Overlay layer (totally or partially). If a feature of the Input layer partially overlaps a feature of the Overlay layer only the portions outside the Overlay layer features are retained. We will use it to filter out all the natural land use leaving only the industrial and urbanized

ones. To do so, first, we have to select the natural land use features:

- Right-click on the landuse\_dissolved layer on the Layers panel
- Open its Attribute table
- Manually select the features having natural land use classes (like the following ones) by holding Ctrl and clicking on the row number on the left
  - heath
  - forest
  - scrub
  - orchard
  - natural\_reserve
  - farm
  - meadow
  - vineyard
  - grass

G	landuse_dissolve	d :: Features Total:	19, Filtered: 19, Sel	ected: 9 —	
/	🕱 📑 😂 i 📅 i	i 🖂 🖻 🖹 i 🗞	🗏 💊 🔩 🍸 🗷	🏶 💭 i 🖪 🖩 🖽	1 🚍 । 🗐 🍳
	osm_id	code	fclass	name	
1	38141486	7218	grass	만섥공ì∍É	
2	38141719	7204	industrial	KT&G	
3	363531897	7214	quarry		
4	472004967	7219	heath	í™″성 ê³ ì •ë¦¬	
5	41943278	7213	military	ì^¬¬> <b>É</b> ê³µí∙-	
6	223871785	7206	cemetery	천ì∙^ê³µì∍Éë¬~	
7	452702676	7210	nature_reserve	ìžÉì—°í∙™ìŠµìž¥	
8	32007357	7201	forest	서ì~¤ë¦‰	
9	26259548	7202	park	ì⊄ë¬~	
10	120027471	7216	vineyard		
11	196319894	7217	scrub		
12	38316679	7205	farm	ë† <b>ì</b> 촌진í <b>¥</b> ¥ì² ì	
13	26628717	7211	recreation_grou	ë¹ì§€ìš′ë <b>Å</b> ™ìž¥	
14	174111566	7207	allotments		
15	26271764	7212	retail		
16	26271759	7203	residential		
17	115698838	7215	orchard		
18	31856777	7209	commercial	ë§ <b>¥</b> ì∍ <b>É</b> ì⊲œìž¥	
	119423303	7208	meadow		

Now that we have selected those features, we can go on with the Difference operation. The input parameters are:

- *Input layer*: the landuse\_a\_clip layer
- *Overlay layer*: the landuse\_dissolved layer, but be sure to check the "Selected features only" checkbox
- *Difference*: the path and the name of the output vector layer. Note that if left empty a temporary layer will be created

Q Difference	
Parameters Log	Difference
Input layer  Input layer  Input layer  Selected features only  Overlay layer  Selected features only  Selected features only  Difference  C:/Users/lawfr/Desktop/QGIS_ex/new_def/difference.shp   Open output file after running algorithm	<ul> <li>overlap, reatures in the Over Input layer features that pai feature(s) in the Overlay lay along those features' bound the portions outside the Over features are retained.</li> <li>Attributes are not modified, properties such as area or I features will be modified by operation. If such properties</li> </ul>
0%	
Run as Batch Process	Run Close

Fig. 3.3.4.1: Difference function window

The result will be a land use layer that only retains the features with industrial and urbanized land use types.

#### 3.3.5 Symmetrical difference

The Symmetrical difference, available at *Vector->Geoprocessing Tools->Symmetrical Difference*, provides a function that extracts the portions of features from both the Input and Overlay layers that do not overlap. The attribute table of the Symmetrical Difference result layer contains original attributes from both the Input and Overlay layers.

If you look at the landuse\_a\_clip layer, you will see it has some portions that overlap with water features:

We will use the Symmetrical difference to obtain a land use layer with no feature that has parts in water bodies. To do so, the input parameters are:

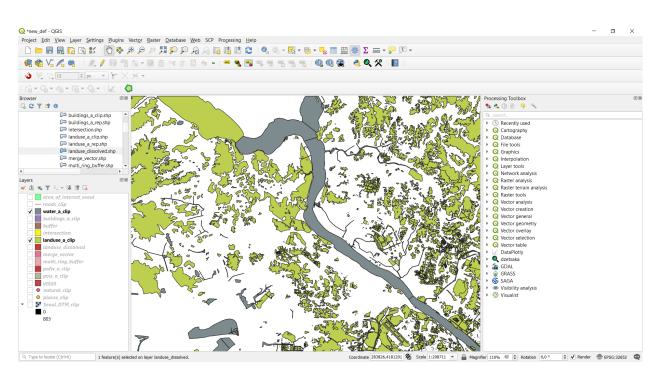


Fig. 3.3.5.1: Water bodies (gray) overlapping landuse features (green)

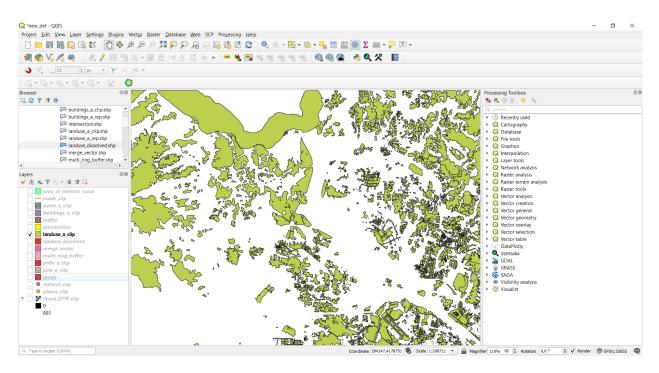


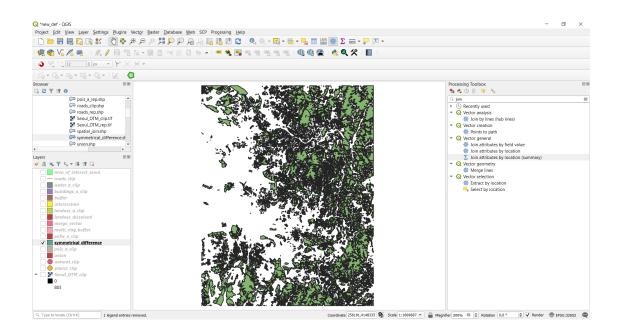
Fig. 3.3.5.2: Only landuse features (green)

- *Input layer*: merge\_vector layer
- Overlay layer: water\_a\_clip layer
- *Symmetrical Difference*: the path and the name of the output vector layer. Note that if left empty a temporary layer will be created

<b>Q</b> Symmetrical Difference		
Parameters Log	Symme	trical dif
Input layer	- This algorith	m extracts the
🖓 merge_vector [EPSG:32652]	features from	n both the Inpu
Selected features only	areas betwe	o not overlap. en the two laye
Overlay layer		e attribute tab Difference lay
water_a_clip [EPSG:32652]		outes from bot
Selected features only	bindrende la	Jero.
Symmetrical difference		
C:/Users/lawfr/Desktop/QGIS_ex/new_def/symmetrical_difference.shp		
✓ Open output file after running algorithm		
0%		
		] [
Run as Batch Process	Run	Close

Fig. 3.3.5.3: Symmetrical difference function window

Once the execution is done, the results should look like this:



### 3.3.6 Spatial join

As you saw with the *Intersection* function we can find the buildings that are also religious places. We will now use the Join Attributes by Location function (generally known as Spatial Join) available at *Vector->Data Management Tools->Join Attributes by Location*, to extend the previous analysis. The Join attributes by location is a function that takes an input vector layer and creates a new vector layer that is an extended version of the input one, with additional attributes in its attribute table. This additional attributes and their values are taken from a second vector layer applying spatial criteria to select the values from the second layer that are added to each feature from the first layer in the resulting one. In this way we could add the attribute that specifies the religion practised in a particular building; the input parameters are:

- Input layer: the buildings\_a\_clip layer
- *Join layer*: the pofw\_a\_clip layer
- Geometric predicate: within
- Fields to add: click on the icon on the left and select the "fclass" attribute
- Join type: take attributes of the first located feature only (one-to-one)
- Tick the "Discard record which cannot be joined" option
- *Joined layer*: the path and the name of the output vector layer. Note that if left empty a temporary layer will be created

If you now look at the attribute table of the newly created layer, you will see a new attribute (fclass\_2) that represents the religion practiced in that building.

Q Join Attributes by Location	×
Parameters Log	<sup>4</sup> Join attributes by
Input layer	location
Duildings_a_clip [EPSG:32652]	This algorithm takes an input vector layer
Selected features only	and creates a new vector layer that is an extended version of the input one, with
Join layer	additional attributes in its attribute table.
pofw_a_clip [EPSG:32652]	The additional attributes and their values are taken from a second vector layer. A
Selected features only	spatial criteria is applied to select the
Geometric predicate	values from the second layer that are added to each feature from the first layer
intersects overlaps	in the resulting one.
contains 🗸 within	
equals crosses	
touches	
Fields to add (leave empty to use all fields) [optional]	
0 elements selected	
Join type	
Create separate feature for each located feature (one-to-many)	
✓ Discard records which could not be joined	Q Multiple selection
Joined field prefix [optional]	
	osm_id
Joined layer	✓ fclass
C:/Users/lawfr/Desktop/QGIS_ex/new_def/spatial_join.shp	name
<ul> <li>Open output file after running algorithm</li> </ul>	
Unjoinable features from first layer	
[Skip output]	
Open output file after running algorithm	
0%	Cancel
Run as Batch Process	Run Close Help
Null do battin FTUCESS	Kull Close Help

Fig. 3.3.6.1: Join Attributes by Location function window

Q	Joined layer :: Featu	ıres Total: 1020, Filt	ered: 1020, Sele	cted: 0			-	×
/	2 B 2 6 i	× 🖻 🖻 I 🗧 🗧	💊 🔩 🍸 🗷	🔖 🗭 🐻 🏙	= I 🗇 🔍			
	osm_id	code	fclass	name	type	fclass_2		-
46	605764872	1500	building	ì∱ˆë′‰ì²œêµ <b>É</b> í	church	christian		
47	578142377	1500	building	효성ì"		buddhist		
48	578142376	1500	building			buddhist		
49	578142379	1500	building	ê′€ <b>ì¥</b> Œì "		buddhist		
50	578142378	1500	building	지장ì "		buddhist		
51	581469040	1500	building	ì< <b>íÅ</b> ‰ìœì¥¼ê		christian_metho		
52	579134045	1500	building	ì~ <b>ü</b> ſ™″êµ <b>É</b> 회		christian		
53	583862858	1500	building	ì<림铱대ê		christian_catholic		
54	581730462	1500	building	별ë, ẽř성ë<¹		christian_catholic		
55	578142369	1500	building	천ë¶^ì "		buddhist		
56	578142368	1500	building	대ì>ë³î "	church	buddhist		
57	578142371	1500	building	ë§Œì~~리ì⊲¤		buddhist		
58	578142370	1500	building	ë²∙ê³ ê° <b>ü</b>		buddhist		
59	578142373	1500	building	천ë³′ë£∵		buddhist		
60	578142372	1500	building	ë,~îœ íf€ë£Œ		buddhist		
61	578142375	1500	building	ë¶≏쥌ê°ü		buddhist		
62	578142374	1500	building	서림ë<¹		buddhist		
<b>T</b>	Show All Features	4500	le collection en	a and U h		ala ni aki a n		-

**Note:** The Join Attributes by Location function, specifying the "intersects" predicate, also translates the "Identity" function of ArcGIS

## 3.4 Proximity analysis

We will now focus on operations involving point layers. Please note that QGIS, when clipping a Point layer, also converts its geometry type to MultiPoint. This geometry type is not suitable for some of the functions we will use in the next step and therefore we will first see how to convert the geometry type back to Point.

### 3.4.1 Convert geometry type

Available at *Processing Toolbox->Vector geometry->Convert geometry type*, it provides an algorithm that allows to convert the MultiPoint features to single Point features. To do so, the input parameters are:

- *Input layer*: the point layer whose geometry we want to convert. In this example we use the places\_clip layer
- New geometry type: select Centroids
- *Converted*: the path and the name of the output vector layer. Note that if left empty a temporary layer will be created

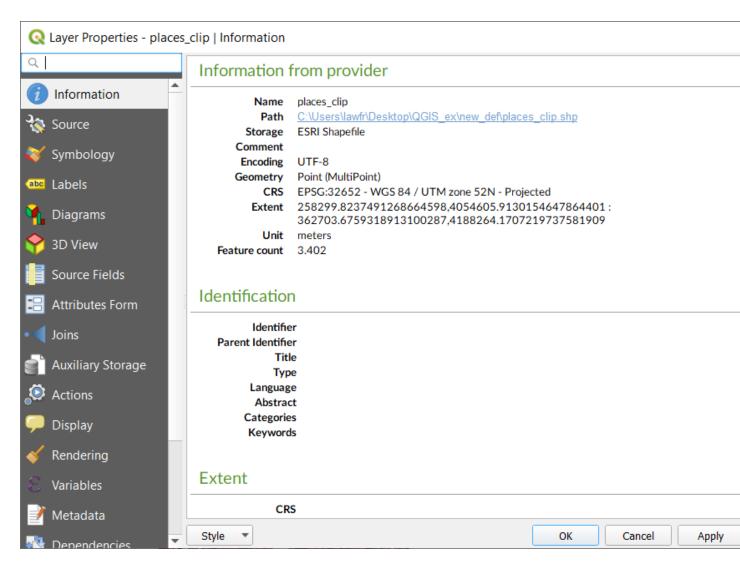


Fig. 3.4.1: In the information of the places\_clip layer we can see that its geometry is MultiPoint

S Convert Geometry Type		
Parameters Log	•	Convert geomet
Input layer           * * places_clip [EPSG:32652]	• 🦻	This algorithm generates a based on an existing one, w type of geometry.
Selected features only New geometry type		Not all conversions are poss instance, a line layer can be point layer, but a point layer
Centroids	•	converted to a line layer.
Converted C:/Users/lawfr/Desktop/QGIS_ex/new_def/places_clip_point.shp		See the "Polygonize" or "Lin- algorithm for alternative opt
✓ Open output file after running algorithm		
0%		
Run as Batch Process		Run Close

Fig. 3.4.1.1: Convert geometry function window

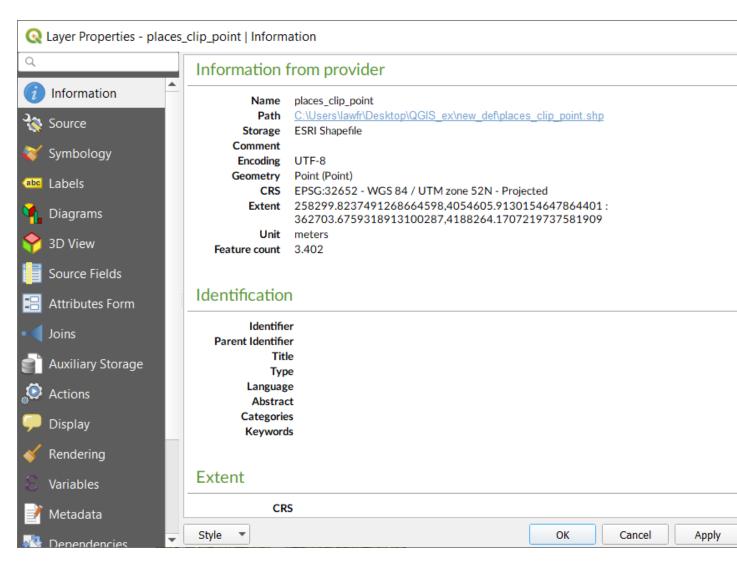


Fig. 3.4.1.2: Now the places\_clip layer's geometry is MultiPoint

In order to continue with the following functions, please convert the geometry type also for the natural\_clip point layer.

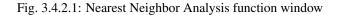
**Note:** After you are done with the conversion, you can remove the previous point layers and include in the project only the new ones.

### 3.4.2 Average Nearest Neighbor

Available at *Processing Toolbox->Vector Analysis->Nearest Neighbour Analysis*, it provides a function that performs nearest neighbor analysis for a point layer. The output is generated as an HTML file with the computed statistics. We perform the Nearest Neighbor Analysis with the natural\_clip\_point point layer; the input parameters are:

- *Input layer*: the natural\_clip\_point layer
- *Nearest Neighbor*: the path and the name of the output HTML file. Note that if left empty a temporary file will be created

🔇 Nearest Neighbour Analysis		
Parameters Log	•	Nearest neighbo analysis
Input layer          * natural_clip_point [EPSG:32652]       *        ?		This algorithm performs nea
Selected features only		analysis for a point layer. Output is generated as an h
Nearest neighbour         C:/Users/lawfr/Desktop/QGIS_ex/new_def/nearest_neighbor_analysis.html		the computed statistical value
0%		
Run as Batch Process		Run Close



Once the operation is done, you can open the HTML file containing the results and you will see information about the Observed mean distance, Expected mean distance, Nearest neighbour index, Number of points, and Z-Score. S nearest\_neighbor\_analysis.html 🛛 🗙

 $\rightarrow$  C

③ Archivio | C:/Users/lawfr/Desktop/QGIS\_ex/new\_def/nearest\_neighbor\_analysis.html

+

Observed mean distance: 361.71124665079316

Expected mean distance: 1392.924296651822

Nearest neighbour index: 0.25967760596913986

Number of points: 1764

Z-Score: -59.48412256905442

Fig. 3.4.2.2: The HTML result when opened in the browser

We now see how to calculate the nearest feature to a given point or set of points in QGIS. We distinguish between distances from point to point, and from point to a line or polygon layer.

### 3.4.3 Distance from point to point

Available at *Processing toolbox->Vector analysis->Distance to nearest hub (points)*, it provides an algorithm that computes the distance between point features taken as the origin and their closest point destination. In this case, we will calculate the distance from the places\_clip\_point layer to the natural\_clip\_point layer. The input parameters are:

- Source points layer: the places\_clip\_point shapefile
- Destination hubs layer: the natural\_clip\_point shapefile
- Hub layer name attribute: osm\_id
- Measurement units: meters
- *Hub distance*: the path and the name of the output vector layer. Note that if left empty a temporary layer will be created

The result is a copy of the places layer, but each point feature has two additional attributes: the id of the nearest natural point feature (HubName) and the distance from it (HubDist), as you can see from its attribute table:

Q Distance to Nearest Hub (Points)	
Parameters Log	Distance to near
Source points layer	(points)
<pre>places_clip_point [EPSG:32652]</pre> ▼ …	Given an origin and a destin
Selected features only	this algorithm computes the between origin features and
Destination hubs layer	destination one. Distance ca based on the features center
* natural_clip_point [EPSG:32652]	The resulting layer contains
Selected features only	center point with an addition indicating the identifier of th
Hub layer name attribute	destination feature and the
abc osm_id	
Measurement unit	
Meters	
Hub distance	
C:/Users/lawfr/Desktop/QGIS_ex/new_def/distance_points.shp	
✓ Open output file after running algorithm	
0%	
Run as Batch Process	Run Close

Fig. 3.4.3.1: Distance from point to point function window

/		- 1 1 1 5	🔊 💊 🕇 🔳	🔖 🔎 i 🖺 📗 i	= I 🗊 🍭				
	osm_id	code	fclass	population	name	HubName	HubDist		
1	415152608	1003	village	0	???	5750034513	3989,27636557		
2	415152609	1003	village	0	???	5750034513	2629,84114765		
3	415152605	1003	village	0	???	2379704396	2983,39820915		
ļ	415152607	1003	village	0	???	2379704396	5290,58934587		
5	415152612	1003	village	0	???	443204531	1308,62028067		
5	415152613	1003	village	0	???	443204531	3412,43746987		
7	415152610	1002	town	0	??	5750034513	3473,72840239		
3	415152611	1003	village	0	???	5750034513	2941,26360602		
9	415152616	1003	village	0	???	5750034513	7732,49995462		
10	415152619	1003	village	0	??	5750034513	6452,27435511		
11	415152614	1003	village	0	???	443204531	6048,86114397		
12	415152615	1003	village	0	???	443204531	6075,50970225		
13	415152745	1003	village	0	???	4734689121	2460,75296840		
14	415152746	1003	village	0	???	4734689121	4239,22913044		
15	415152739	1003	village	0	???	443204531	11914,4757320		
16	415152743	1003	village	0	???	443204531	9433,21282902		
17	415152749	1003	village	0	???	4734689121	5595,30128481		
18	415152750	1003	village	0	???	5624510429	7759,01492254		

### 3.4.4 Distance from point to layer

Available at *Processing Toolbox->Vector analysis->Distance to nearest hub (line to hub)*, it provides an algorithm that computes the distance between point features taken as origin and their closest destination line or polygon feature.

Note: Distance calculations are based on the centroid of the line or polygon features.

In this case, we calculate the closest forest to each place. To do so, we select all the forests from the landuse\_a\_clip layer:

- Right-click on the landuse\_a\_clip layer in the Layers panel and click on Open attribute table
- Click on the Select features using an expression button
- In the window, write the following expression: "fclass" is 'forest', and then click "Select features"

	osm_id	code	features using an e	expression	
1	26259548	7202		??	
2	26268329	7202	park	?????????	Select by Expression - landuse_a_clip
3	26271759	7203	residential		Expression Function Editor
4	26271764	7212	retail		"fclass" IS 'forest'
5	26271767	7203	residential		Conversions     Date and Time
6	26271768	7203	residential		<ul> <li>Fields and Values</li> <li>Fuzzy Matching</li> <li>General</li> </ul>
7	26335950	7212	retail		Geometry     Map Layers     Maps
8	26628717	7211	recreation_grou	?????	<ul> <li>Math</li> <li>&gt; Operators</li> </ul>
9	26658458	7202	park	???	Rasters     Output preview: 0     Record and Attributes
10	27408452	7202	park	????	Help
11	31564854	7202	park	????	
12	31565064	7203	residential	?????	

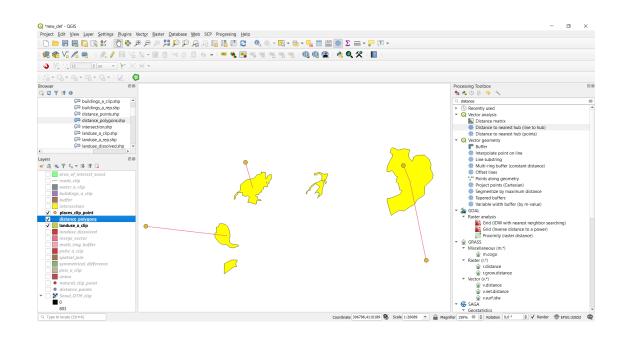
Once we have selected all the forests from the landuse\_clip layer, we can run the Distance to nearest hub (line to hub) function. The input parameters are:

- Source points layer: the places\_clip\_point layer
- Destination hubs layer: the landuse\_a\_clip layer, considering only selected features
- Hub layer name attribute: osm\_id
- Measurement units: meters
- *Hub distance*: the path and the name of the output vector layer. Note that if left empty a temporary layer will be created

The result is a line layer representing all the distances from each places point feature to the nearest forest feature.

🔇 Distance to Nearest Hub (Line to Hub)		
Parameters Log	•	Distance to nea
Source points layer		(line to hub)
📌 places_clip_point [EPSG:32652] 🔹 🛄 🥥		Given an origin and a destir
Selected features only		this algorithm computes the between origin features and
Destination hubs layer		destination one. Distance ca based on the features center
□ Ianduse_a_clip [EPSG:32652]		The resulting layer contains
Selected features only		each origin point with its ne
Hub layer name attribute		destination feature.
abc osm_id		
Measurement unit		
Meters		
Hub distance		
C:/Users/lawfr/Desktop/QGIS_ex/new_def/distance_polygons.shp		
✓ Open output file after running algorithm		
0%		
Run as Batch Process		Run Close

Fig. 3.4.4.1: Distance from point to layer function window



### 3.5 Network analysis

### 3.5.1 Shortest path

In order to find the shortest or fastest path in QGIS, we can use the Shortest path function. Having a layer representing a network, we can calculate the shortest path between two chosen points on the map (point to point), from a point layer to a chosen end point (layer to point) or from a chosen start point to a point layer (point to layer).

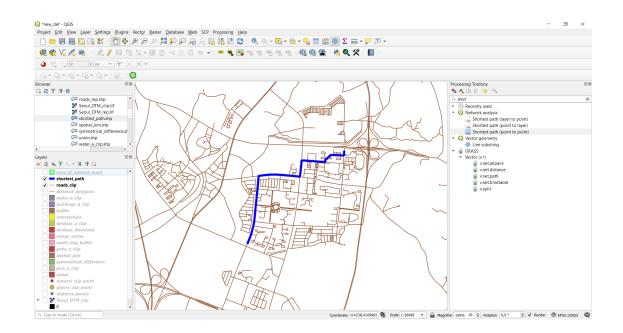
We illustrate here only the point to point option for the sake of computation, but the others are easily deduced from the following example. The function is available at *Processing Toolbox->Network analysis-Shortest path (point to point)*, and the input parameters are:

- Vector network layer: the roads\_clip layer
- *Path type to calculate:* "shortest" (you can also calculate the fastest path given a network layer with maximum velocity information)
- Start point: click on the icon on the right, then choose a starting point from the map
- End point: click on the icon on the right, then choose an ending point from the map
- *Shortest path*: the path and the name of the output vector layer. Note that if left empty a temporary layer will be created

As you can see this algorithm creates a new line layer that represents the shortest path in the provided network to go from the starting point to the end point:

Q Shortest Path (Point to Point)			
Parameters Log		•	Shortest path (p
Vector layer representing network			point)
√° roads_clip [EPSG:32652]	🔊		This algorithm computes op
Selected features only			or fastest) route between gi end points.
Path type to calculate			
Shortest	•		
Start point (x, y)			
314685.5283574448,4110645.667601278 [EPSG:32652]			
End point (x, y)			
315911.4440157223,4111801.636119894 [EPSG:32652]			
Advanced parameters			
Shortest path			
C:/Users/lawfr/Desktop/QGIS_ex/new_def/shortest_path.shp			
✔ Open output file after running algorithm			
0%			
Run as Batch Process			Run Close
Nan do bacen nocessar			Close

Fig. 3.5.1.1: Shortest path (point to point) function window



### 3.5.2 Generate service area

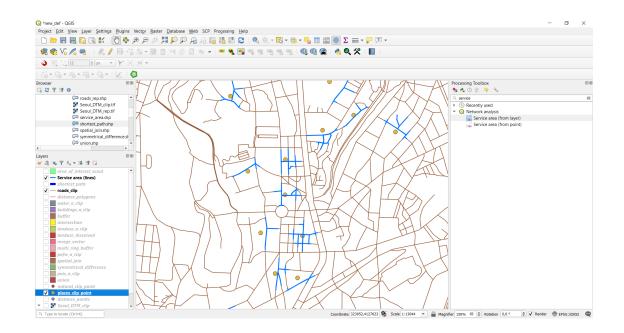
In QGIS, we can also generate a service area using the Service Area function. To perform such, you can search for Service area in the Processing toolbox searchbar and select *Service area (from point)*. This function allows creating a vector with all the parts of a network layer that can be reached within a distance or a time, starting from a point chosen on the map. The same can be done starting from a point layer using *Service area (from layer)*. We will use the last one to calculate the service area for all the places points with a maximum travel distance of 100 meters. The input parameters are:

- Vector network layer: the roads\_clip network
- Vector layer with start points: the places\_clip\_point layer
- *Path type to calculate*: "shortest" (you can also calculate the fastest path given a network layer with maximum velocity information)
- Travel cost: 100 (so that the maximum travel distance is 100m)
- *Service area*: the path and the name of the output vector layer. Note that if left empty a temporary layer will be created

The result will highlight the service area for each point of the places layer:

🔇 Service Area (From Layer)		
	4	
Parameters Log	`	Service area (fr
Vector layer representing network		layer)
▼ 4	>	This algorithm creates a new
Selected features only		all the edges or parts of edg network line layer that can be
Vector layer with start points		within a distance or a time, features of a point layer.
places_clip_point [EPSG:32652]		The distance and the time (I
Selected features only		to as "travel cost") must be respectively in the network l
Path type to calculate		seconds.
Shortest		
Travel cost (distance for "Shortest", time for "Fastest")		
100,000000	\$	
Advanced parameters		
Service area (lines)		
C:/Users/lawfr/Desktop/QGIS_ex/new_def/service_area.shp		
✓ Open output file after running algorithm		
Service area (boundary nodes)		
[Skip output]		
Open output file after running algorithm		
0%		
Run as Batch Process		Run Close

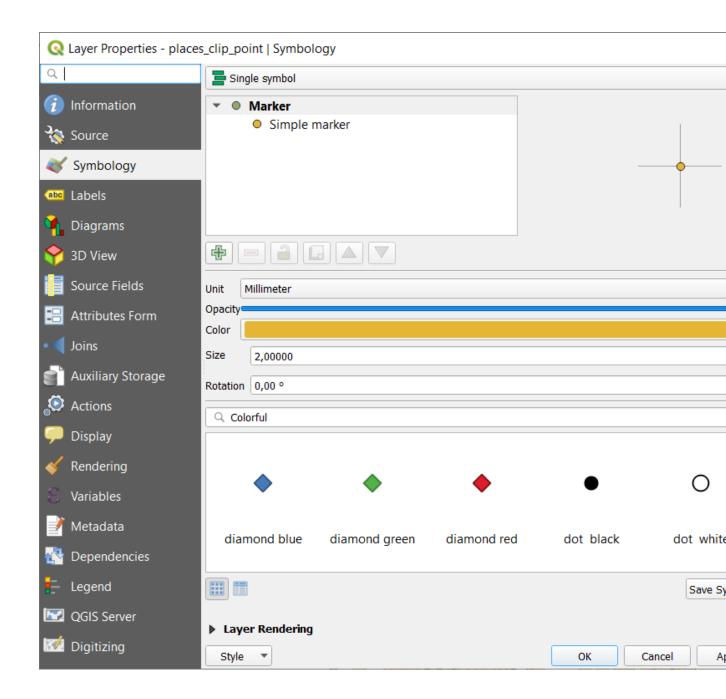
Fig. 3.5.2.1: Generate service area function window



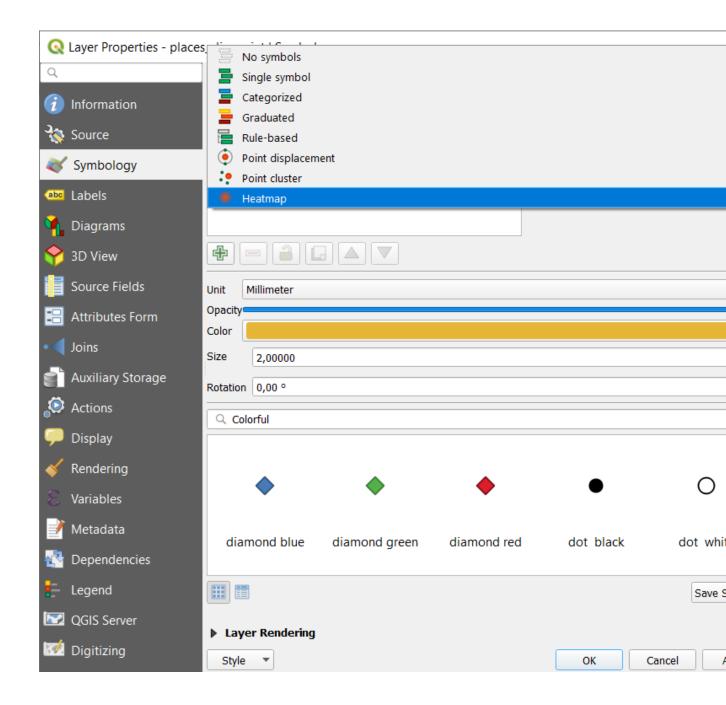
## 3.6 Kernel density

A useful tool to visualize the density of a point layer it is the Heatmap. In QGIS, we can directly use a styling option of the layer.

- Right-click the places\_clip layer on the Layers panel
- Select Properties
- In the menu on the left, select "Symbology"



• In the dropdown menu on top, select Heatmap



- Now you only have to choose the parameters:
  - Color ramp: reds
  - Radius and radius unit of measure: 10 millimetres
  - Maximum value (leave Automatic)

Q Layer Properties - plac	es_clip_point   Syr	nbology			
۹	📕 Heatmap				
(i) Information	Color ramp				
🍓 Source	Radius	10,000000			4
😻 Symbology	Maximum value	Automatic			
(abc) Labels	Weight points by				
🐪 Diagrams	Rendering quality	Best	,	I	
幹 3D View					
Source Fields					
🔡 Attributes Form					
• ┥ Joins					
📄 Auxiliary Storage					
Sections					
🧭 Display					
🞸 Rendering	Layer Rende	ering			
🗧 Variables	Style V		ОК	Cancel	A

• Then click OK button on the bottom

The result is the visualization of the places layer as a heatmap:

### 3.7 Inverse Distance Weighting

We can also interpolate a point layer to create a raster layer out of it. One method to do so is the Inverse Distance Weighting (IDW) interpolation technique. To perform IDW interpolation in QGIS we have to rely on the external functionalities provided by GRASS GIS that are directly accessible form the Processing Toolbox of QGIS. The function is called *v.surf.idw*, and it's available at *Processing Toolbox->GRASS->Vector(.v\*)*. We can compute a raster by interpolating the places\_clip\_point layer, using as the interpolation variable the population. But first, we need to select all the cities and towns that have a valid population attribute (i.e. > 0): to do so, follow these steps:

- Right click on the places\_clip\_point layer and open its Attribute table
- Click on "Select features using an expression"
- In the window, write the following expression: "population" > 0, and then click "Select features"

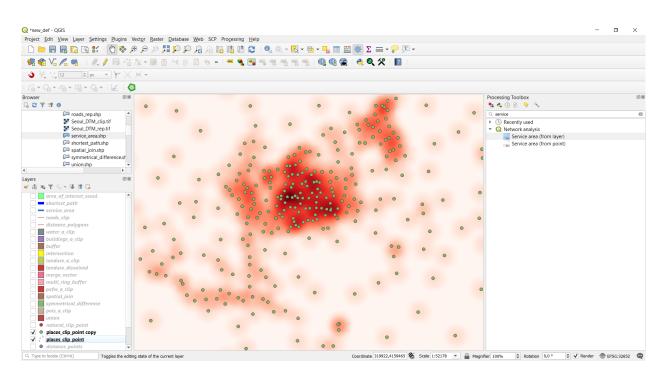


Fig. 3.6.1: Comparison between the heatmap and a copy of the places\_clip layer with the point symbology

	osm_id	code Sel	ect features using a	n expression pn	name				
1	257363469	1001		967510	???				
2	1912365631	1005	national_capital	9631482	??				
3	1912558038	1001	city	849361	???		Select by Expression - places_clip_point		
4	415156263	1001	city	696587	???		Expression Function Editor	Q. Search	Show
5	415153247	1002	town	68754	???		= + - / * ^ II ( ) "\n' "population" > 0	<ul> <li>Aggregates</li> <li>Arrays</li> </ul>	
5	257372341	1001	city	677710	???			Color     Conditionals	
7	1910974426	1002	town	66778	???			<ul> <li>Conversions</li> <li>Date and Time</li> <li>Fields and Values</li> </ul>	
3	415157745	1002	town	66685	???			NULL abc osm_id	
)	415158162	1001	city	631531	???			123 code abc fclass	
10	415159708	1002	town	60882	???	0	Output preview: 1	123 population abc name	
1	1912558035	1001	city	587764	???		Help		
2	1912558042	1002	town	527600	???				

Now that we have selected all the places with population greater than 0, we can proceed with the IDW interpolation. Click on the v.surf.idw function and input the following parameters:

- *Input vector layer*: the places\_clip\_point layer, be sure to tick the "Selected features only" option
- Number of interpolation points: 200

- Attribute table column with values to interpolate: select "population"
- *Interpolated IDW*: the path and the name of the output raster layer. Note that if left empty a temporary layer will be created

Parameters Log		4	v.surf.idw
Input vector layer			Surface interpolation
° places_clip_point [EPSG:32652]	🦻		data by Inverse Dista Weighting.
✓ Selected features only			, , , , , , , , , , , , , , , , , , ,
Number of interpolation points [optional]			
200			
Power parameter; greater values assign greater influence to closer points [optional]			
2,000000			
Attribute table column with values to interpolate			
123 population	•		
Don't index points by raster cell			
Advanced parameters			
Interpolated IDW			
C:/Users/lawfr/Desktop/QGIS_ex/new_def/IDW.tif			
✓ Open output file after running algorithm			
0%			

The result should initally look like this:

roject Edit View Layer Settings Plugins Vector Baster Database Web SCP Processing		
🗅 🖶 🗟 🖓 유 🤤 🔍 속 🖉 🐘 🖾 🖓 🖓 🔚 🔚	😂 🛛 🍳 - 🔣 - 📑 - 🧏 🛅 📓 🎆 Σ 🚍 - 🍃 💷 -	
🔍 🎕 Vi 🔏 🖏    //. / 🗒 🕆 友 - 🕺 🗇 🗟 🖉 🤣 I 🛥 🍕	📇 👒 🤫 🦏 🧠 । 🕼 🦓 😭 । 👌 🔍 🛠 । 🔝 ।	
3 🐙 😳 12 🔶 px 👻 Y 🗙 X 👻		
G • Q • Q • Q • Q • 🖌 🔕		
owser 🛛 🕅		Processing Toolbox
		- 🎭 🍓 🕓 🖹 i 🦻 i 😽
☆ Favorites		Q idw 6
C:\Users\lawfr\Documents\Politecnico\2_1\GIS		<ul> <li>Recently used</li> </ul>
C:\Users\lawfr\Documents\Politecnico\2_2\Geoinform;     Project Home		v.surf.idw
Hone		Interpolation
		T GDAL
🍄 GeoPackage		<ul> <li>Raster analysis</li> </ul>
/ SpatiaLite		Grid (IDW with nearest neighbor searching)
		▼
yers 2018 * (최 · 종, 후 · 등 · · · · · · · · · · · · · · · · ·		
		w r.suf.idw
area_of_interest_seoul		✓ Vector (v.*)
water_a_clip		🎡 v.surf.idw
buildings_a_clip		
landuse_a_clip		
pofw_a_clip		
pois_a_clip		
V Interpolated IDW 46867.3		
9,45956e+06		
places_clip_point		
natural_clip_point		
Seoul_DTM_clip		
0		
803		

Since this result is not really optimal for an analysis, we can try to style it in a better way. To do so:

- Right-click on the newly created raster and click on "Properties"
- In the window, select "Symbology" from the menu on the left
- Change the first option ("Render type") to "Singleband pseudocolor"
- Now change the classification mode to "Quantile"
- Change the number of classes to 9
- Click "Ok"

	Band Rendering						
i Information	Render type Singleb	and pseudo	color 🔻				
Source	Band		Band 1 (G	ray)			•
💕 Symbology	Min		46867.3		Max	9.45956e+	•06
Transparency	🕨 Min / Max Valu	e Settings					
🖂 Histogram	Interpolation			Linear			•
🖌 Rendering	Color ramp						
Pyramids	Label unit suffix						
📝 Metadata	Value	Color	Label				<b>A</b>
Legend	46867.2913		46867	.2913706042			
🗹 QGIS Server	1270517.90		12705	17.90074457			
	2494168.51		24941	68.51011854			
	3717819.11		37178	19.11949251			*
	Mode Continuous	•					Classes 5
inuous 🔺	Classify 🕀		2				
l Interval	Clip out of range	e values					
ntile	▼ Color Rendering						
	Blending mode Norr	nal			<b>*</b>		👆 Reset
	brending mode mon						1 House

Layer Properties - I	nterpolated IDW   Syr	nbology				
	<ul> <li>Band Rendering</li> </ul>					
Information	Render type Singleb	and pseudo	ocolor 🔻			
Source	Band		Band 1 (Gray)			
Symbology	Min		46867.3	Max		9.45956e+06
ransparency	🕨 Min / Max Valu	e Settings				
togram	Interpolation		Linear			
dering	Color ramp					
amids	Label unit suffix					
data	Value	Color	Label			
ld	46867.2913		46867.2913706042			
Server	338660.898		338660.89822132			
	404549.777		404549.77718761			
	470438.656		470438.656153901			
	Mode Quantile Classify Clip out of range Clip out of range Clip out of range Blending mode Norm Brightness Saturation	e values		Contrast		
	Style 🔻				ОК	Cancel A

Now the result should be subdivided in 9 classes that contain an equal number of pixels each; the raster should look like this:

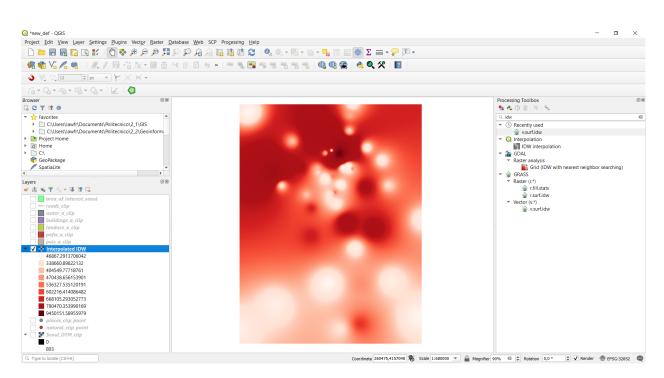


Fig. 3.7.1: Now it's easier to distinguish the variation in population: note for example the dark red area in the top part of the image that represents the area where the capital Seoul is.

# CHAPTER 4

### **Raster operations**

### 4.1 Merge raster

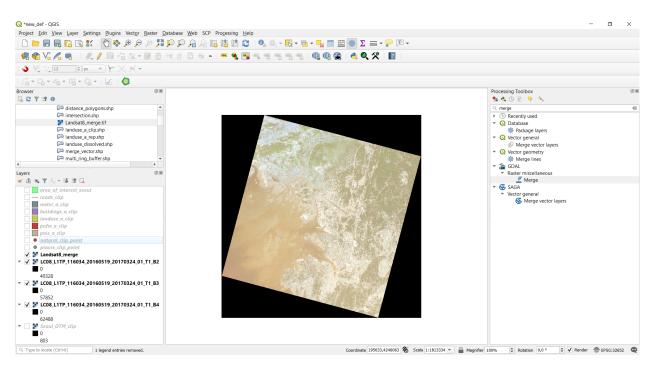
We see now how to Merge multiple raster layers. We do it by combining the Landsat8 imagery bands with the aim of creating a multi-band raster. To do so, we have to first add the raster bands into our QGIS project. From the the Browser panel, navigate to the raster data folder and into the folder containing the Landsat data (Landsat8\_20160519\_20170324\_01\_T1). Then add the raster ending with B2, B3 and B4 (in Landsat images they represent respectively the blue, green and red channels). These images are already in our project CRS (*EPSG:32652 - WGS 84 / UTM zone 52N*) so once you added the raster data, go directly to *Raster->Miscellaneous->Merge*, and put this input parameters:

- Input layers: click on the left symbol and select the Landsat B2, Landsat B3 and Landsat B4 images (LC08\_L1TP\_116034\_20160519\_20170324\_01\_T1\_B2. LC08\_L1TP\_116034\_20160519\_20170324\_01\_T1\_B3.TIF TIF. and LC08\_L1TP\_116034\_20160519\_20170324\_01\_T1\_B4.TIF)
- Tick the "Place each input file into a separate band" checkbox: when not activated, this option is used to merge raster layers covering different area and not overlapping between each other. In our case we want to merge images that totally overlap between one another
- Output data type: change it to UInt16 to be consistent with the original rasters
- *Merged*: the path and the name of the output raster layer. Note that if left empty a temporary layer will be created

The obtained multiband raster at first will look like this:

Q Merge       Parameters       Log       Input layers	×
0 elements selected Grab pseudocolor table from first layer ✓ Place each input file into a separate band Output data type UInt16	
Advanced parameters Merged C:/Users/lawfr/Desktop/QGIS_ex/new_def/Landsat8_merge.tif      Open output file after running algorithm	<ul> <li>✓ Multiple selection</li> <li>✓ LC08_L1TP_116034_20160519_20170324_01_T1_B2 [EPSG:32652]</li> <li>✓ LC08_L1TP_116034_20160519_20170324_01_T1_B3 [EPSG:32652]</li> <li>✓ LC08_L1TP_116034_20160519_20170324_01_T1_B4 [EPSG:32652]</li> </ul>
GDAL/OGR console call python3 -m gdal_merge -separate -ot UInt16 -of GTiff -o C:/Users/lawfr/Desktop/QGIS_ex/ processing_eb9454dd498d4b91a46856d4fc8720a6/6b62b2fb9e0548cbafc8f98874d08613/m	
0%	Cancel
Run as Batch Process	Run Close Help

Fig. 4.1.1: The merge raster function window



To obtain what is called a "true color image", so a multiband raster whose color resembles the real ones, you have to change the band assigned to each color. To do so, right-click on the Landsat-merge raster and select "Properties". Go in the "Symbology" section, and you will see a window like the following:

Q Layer Properties - La	andsat8_merge	e   Symbology			
۹	▼ Band Rende	ering			
information	Render type	Multiband color 🔹	7		
💸 Source	Red band	Band 1 (Gray)			
😻 Symbology		Min 0		Max 12260	
I Transparency	Green band	Band 2			
 Histogram		Min 0		Max 12207	
Kendering	Blue band	Band 3			
		Min 0		Max 13123	
Pyramids	Contrast enhancement	Stretch to MinMax			
📝 Metadata		x Value Settings			
E Legend		•			
QGIS Server	<ul> <li>Color Render</li> </ul>	_			
	Blending mode	Normal	<b>•</b>		•
	Brightness			ntrast	0
	Saturation		0 🗘 Gra	ayscale Off	
	Hue	Colorize	<ul> <li>Strength</li> </ul>		
	▼ Resampling				
	Zoomed: in	Nearest neighbour 💌 out	Nearest neighbour 💌 Ove	ersampling 2,00 🜲	
		Thumbnail	Legend	Palette	
		and the second second			
	Style 🔻			ОК	Cancel Apply

Then, assign to the Red band the Band 3, to the Green band the Band 2 and to the Blue band the Band 1.

Q	Layer Properties -	Landsat8_merge	Symbology			
Q		Band Rende	ering			
i	Information		Multiband color 🔹	]		
ે્	Source	Red band	Band 3			
~	Symbology		Min 0		Max 13123	
	Transparency	Green band	Band 2			
	Histogram		Min 0		Max 12207	
~	Rendering	Blue band	Band 1 (Gray)			
	-		Min 0		Max 12260	
	Pyramids	Contrast enhancement	Stretch to MinMax			
1	Metadata		x Value Settings			
÷	Legend		5			
	QGIS Server	Color Rende	ering			
		Blending mode	e Normal	•		
		Brightness		0 🗢 🛛 Cor	ntrast	0
		Saturation		0 🔷 Gra	ayscale Off	
		Hue	Colorize	Strength		
		▼ Resampling				
		Zoomed: in	Nearest neighbour 💌 out	Nearest neighbour 🔻 Over	rsampling 2,00 🜲	
			Thumbnail	Legend	Palette	
1		Style 🔻			ОК	Cancel Apply
		- Style				Concer (PPN)

Click "Ok" and the true color map obtained should look like this:

Project Edit View Layer Settings Plugins Vector		delp 3 Ι Φ, Φ, - Ε, - Ε, - Ε, ΙΙΙ ΙΙΙ 🐺 Σ 🛲 - 🌄 II -	
	2 ti × 6 2 5 × 1 = 9.		
👌 🐙 📜 12 🗣 px 🐨 🏲 🗙 🗡			
6-9-2-8-8-9-2 🧔			
rowser	88		Processing Toolbox
C distance_polygons.shp	<b>A</b>		Q merge
intersection.shp		And the second sec	<ul> <li>Recently used</li> </ul>
Landsat8_merge.tif			<ul> <li>Q Database</li> </ul>
Ianduse_a_clip.shp Ianduse a rep.shp			Package layers
Ianduse_a_rep.snp			✓ Q Vector general I Merge vector layers
Rerge_vector.shp		the second of the second second second second	<ul> <li>Werge vector rayers</li> <li>Q Vector geometry</li> </ul>
multi_ring_buffer.shp	-	and the second	A Merge lines
	<b>F</b>		👻 🚋 GDAL
ers	22 (26)		<ul> <li>Raster miscellaneous</li> </ul>
(山) 🔍 🝸 🖏 🐨 💷			Z Merge
area_of_interest_seoul			<ul> <li>SAGA</li> <li>Vector general</li> </ul>
roads_clip			Vector general
water_a_clip			<b>v</b>
buildings_a_clip			
landuse_a_clip			
pois_a_clip			
natural_clip_point			
places_clip_point			
✓ 🐏 Landsat8 merge			
V K LC08_L1TP_116034_20160519_20170324_0	_T1_B2		
0 40328			
V LC08_L1TP_116034_20160519_20170324_0	T1 B3		
57852			
V F LC08_L1TP_116034_20160519_20170324_0	_T1_B4		
0			
62488			
Seoul_DTM_clip			
803			

**Note:** This is not an accurate true color map because there was no atmospheric correction applied to the Landsat images, so you'll find that its colors are brighter than the real ones.

# CHAPTER 5

Index

• genindex

### Index

# A

Average Nearest Neighbor,44

## В

Buffer, 21

## С

Clip raster, 17 Clip vector, 17 Convert geometry type, 40 Create new vector layer, 13

# D

Difference, 32 Dissolve, 29 Distance from point to layer, 47 Distance from point to point, 45

# G

Generate service area, 52

## I

IDW,57 Import raster data,9 Import vector data,7 Intersection,27

## Κ

Kernel density, 54

## Μ

Merge raster,65 Merge vector,25 Multi-ring buffer,23

## R

Reproject raster,11 Reproject vector,10

## S

Shortest path, 50 Spatial join, 38 Symmetrical difference, 35

## U

Union,27