
Noise Mapping: Paper ICA2016-517**Time and cost saving software techniques and their application in large-scale noise mapping projects****Antonio Notario^(a)**^(a) DataKustik GmbH, Germany, antonio.notario@datakustik.com**Abstract**

Noise mapping is one of the most important tools in the global fight against noise. The implementation of the Environmental Directive 2002/49/EC had been the main trigger in the development of techniques and software aiming to handle Noise Mapping projects in large areas or even complete cities. The implementation of an effective Policy against noise comprises several aspects from the introduction of a limiting value system, selection of the calculation methods, project management, up to the availability of results to the public in a clear, comprehensive and accessible way. On the other hand the time needed for and the quality of noise maps and action plans depend a lot on the software tools applied and therefore any modification of data which needs manpower can raise the time and cost enormously. A technique based on scripting is presented for automating time consuming tasks such as correcting models after the import of data from various third party formats, iterating over different configurations of calculations, summing up source-specific noise maps including level corrections or even exporting maps to an Online Noise Map Interface to make results accessible to the public.

Keywords: Noise Mapping

Time and Cost Saving Software Techniques and their application in Large-Scale Noise Mapping Projects

1 Introduction

Noise mapping has become the main tool of the European Noise Policy. The target of this approach is to get reliable data about the noise pollution in the cities and to come to optimal harmonized strategies to achieve the needed noise mitigation. All the noise mapping projects that have been carried out after the first and second rounds have shown that digital city models are the basis of powerful noise information systems used by the city administrations and the public. Any question that arises about the noise load of a certain region can be answered immediately without spending further money for expensive experts.

Calculations are executed by using national or international noise calculation standards; examples are the Interim Methods recommended by the EU which will be substituted by the new CNOSSOS-EU [1] Methods very soon. Either way, the strategic noise assessment in agglomerations includes a number of aspects to consider beyond the calculation techniques. As an example, strategic noise maps must be updated every 5 years in order to include new input data – e.g. new emission data such as traffics – as well as the noise reduction measures that have been implemented as a result of the approved Noise Action Plans during the previous stage. Another important aspect is that according to article 9 of the EC Directive [2] the Member States have to ensure that strategic noise maps and action plans are made available and disseminated to the public.

From the first stage where the digital model is created to the last stage where the noise mapping results are made available to the public, the time needed and therefore the quality of the project depends on the software tools applied. This paper discusses some of the latest developments which support these tasks in an effective way.

2 Data conditioning and creation of the 3D city model

The model of a city is produced by combining different objects such as topography, buildings, screens or industrial and traffic noise sources. These physical objects are usually imported from third party formats such as GIS-Systems while the attributes can be imported from given databases into the software platform. In practice, the availability of data from the environmental authorities is not always granted due to several facts. Firstly, the information is usually dispersed across the different departments and secondly datasets coming from Land-Register information are normally saved as 2D information. These situations lead to the need of collaboration between the authorities and the noise consultants in order to build an accurate digital city model constituting one of the most cost relevant parts of a noise mapping project.

2.1 Import of data and automated conditioning

It is well known that the data conditioning easily represents a minimum of the 60% of the overall project's timeline. This amount of time can be higher as it depends on the type and quality of the input of the imported data. In case no official data formats are available – e.g. GIS databases – or those formats are incomplete, additional import formats may be used to build the final model. One of these formats is the Open Street Maps project [3] consisting of a collaborative platform of enthusiasts for map creation based on GPS data. Contrary to the existing Google Maps and Google Earth formats OSM does not include imagery data but true objects like buildings, roads, railways and even areas of land use that are recognized by the prediction software. Due to the fact that OSM is an open source format which means that basically everyone can contribute, the level of detail as well as the embedded information may vary heavily depending on the region. Nevertheless this format is of great help to improving data from incomplete GIS information. Figure 1 presents the Open Street Map website with a selected section of the city of Buenos Aires which can be exported to a file by using the export panel.

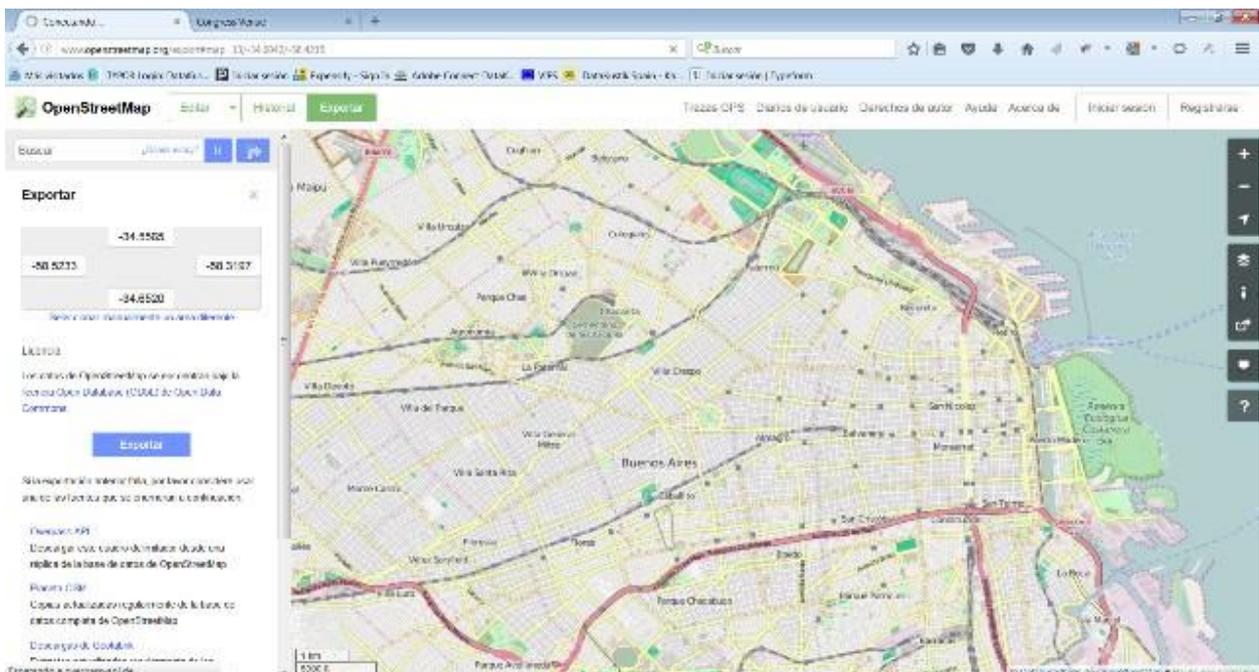


Figure 1: Open Street Map website with export panel using Mozilla Firefox

The downloaded file can then be imported into the prediction software [4] directly by selecting the Open Street Maps (.xml) format. Figure 2 shows that, after the import process, the data is proven to be incomplete. As an example, many buildings do not have a valid height or they are below the ground while underground noise sources - like metro trains or roads through tunnel sections - are imported and therefore they shall be deactivated before the calculation. Even the different noise prediction software products usually implement tools for solving these problems, the situation could get more complicated as the project size becomes bigger and therefore the automation of tasks can be of enormous help in those cases.

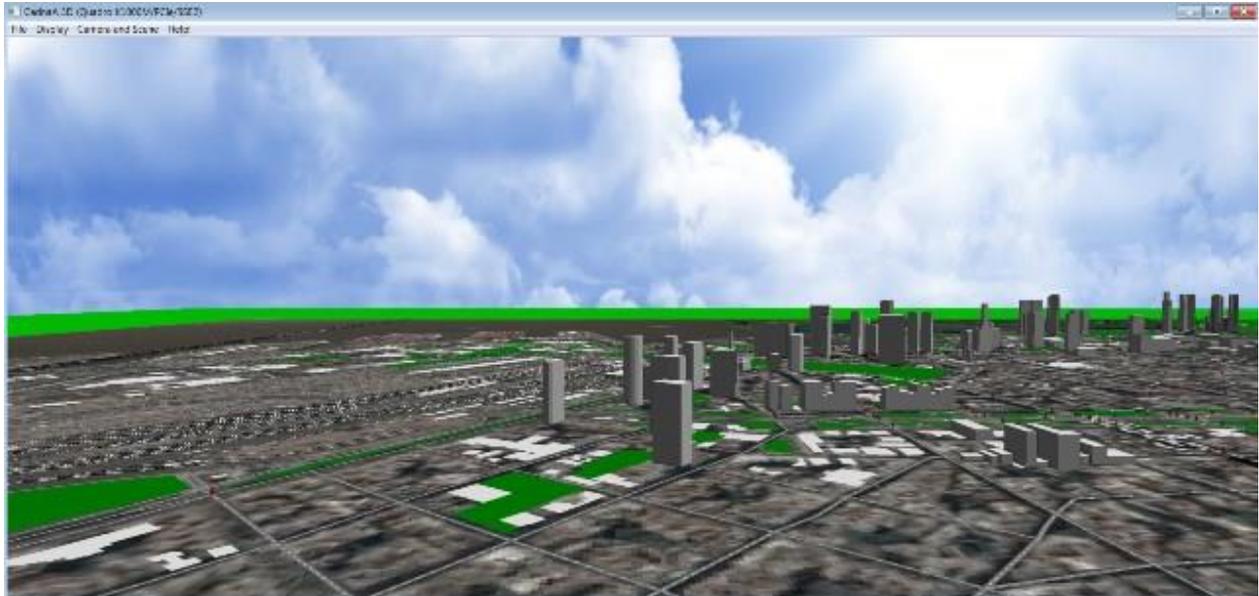


Figure 2: 3D View of the City of Buenos Aires model. Many buildings have incorrect or missing height information

Using the example of the city of Buenos Aires a Lua [5] script can be written and executed directly from the noise prediction software [4]. A portion of the script is presented in Figure 3:

```

1  -- iterates over objects and set height for cleaner display after OSM import
2  -- houses = 10m (if height was 0)
3  -- foliage = 5m (if height was 0)
4  -- road,rail,ppl = 0.1m
5  -- iterates over areas of land use tables and assign an existing area of land use
6  -- also deactivates all railways with memo "tunnel=yes" (subway)
7  cna.print("Postprocessing OSM imported data for display...")
8  for h in cna.tables.Haus:all() do
9      if h.HA < 0.001 then
10         h.HA = 10
11     end
12 end
13 for f in cna.tables.foliage:all() do
14     if f.HA < 0.001 then
15         f.HA = 5
16     end
17 end
18 for t in cna.tables.track:all() do
19     if t.memo_var.tunnel == "yes" then
20         t.MARK = -1
21     else
22         t.HA = 0.1
23     end
24 end
25 for a in cna.tables.ar_ind:all() do
26     if a.memo_var.landuse == "Hospital" then
27         a.ARTI = 1
28     end
29     if a.memo_var.landuse == "residential" then

```

Figure 3: Lua Script example (comments are shown in green color)

The script iterates over the table of buildings, finds those having a height value less than 0.1m and replaces it with any valid height defined by the user –here, 10m-. The script can apply the same procedure for any attribute and any object and therefore the deactivation of underground

sources – which include the attribute “tunnel=yes” - or the correct assignment of the different land uses can be achieved.

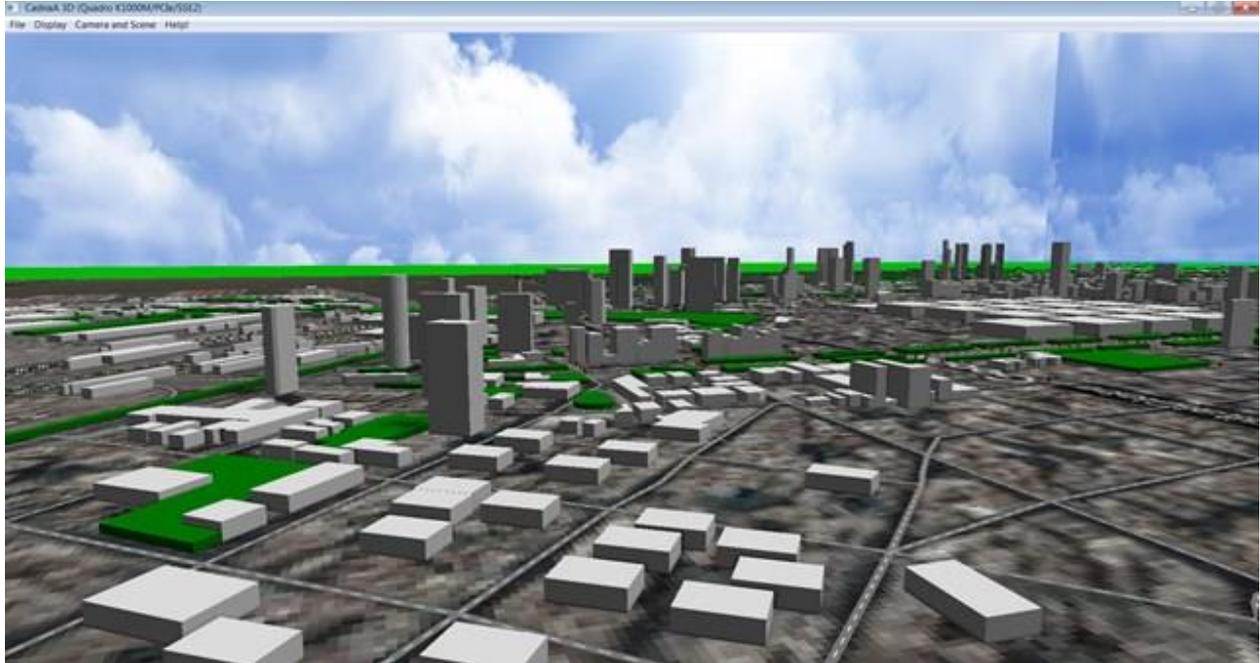


Figure 4: 3D View of the City of Buenos Aires with corrected data after running the Lua script

There are some cases where certain datasets such as the elevation model are sometimes too detailed in comparison with the rest of available data. Even if modern noise prediction software is able to handle and process a huge amount of data, 3D models with a too detailed digital terrain model will increase the calculation time without delivering more accurate calculation results. Thanks to the application of a Lua script these models can be simplified in such a way that the final height deviation stays within a predefined level in meters. In figure 5 a map of terrain heights is displayed, showing the difference between the original dataset and the resulting new terrain after the simplification. All height deviations are below 1m but the calculation time will differ dramatically.

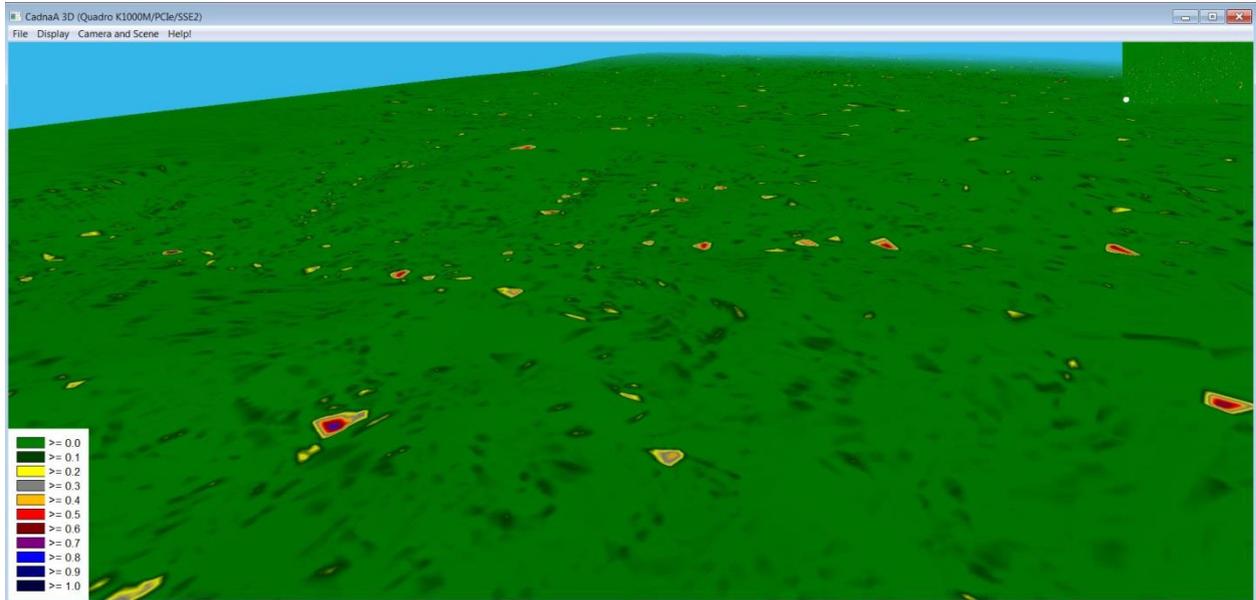


Figure 5: Height difference between the original and simplified elevation models

3 Acceleration techniques and their impact on the calculations

In the scope of noise mapping projects noise the use of highly developed software platforms is extremely recommendable because many additional techniques can be combined with a certain calculation method in order to improve the prediction results. Some of these techniques are known as “acceleration techniques” and are used to reduce calculation time while reasonable results are obtained. In other cases some techniques give more precision although the calculation time is incremented. All the relevant calculation settings with regards to the noise mapping such as the projection of line sources, selection of search radii, grid interpolation or the order of reflection should be balanced in order to optimize the calculation speed versus the reduction of the accuracy of the calculated levels. These uncertainties can be determined automatically with techniques described in the standard DIN 45687 [6].

This feature allows the statistical distribution of a selected number of receiver points all over the calculation area followed by the calculation of noise levels, first with a reference configuration of calculations – e.g. without acceleration techniques - and then with these settings intended for the calculation of the noise map. The level differences at the receiver points are then analyzed and the interval of deviations caused by the differences within configurations is determined. This helps in the decision about whether a more detailed description and calculation of the model will really deliver a benefit regarding the final results. As an example, Figure 6 shows the statistical results and level differences when using two and three order of reflections.

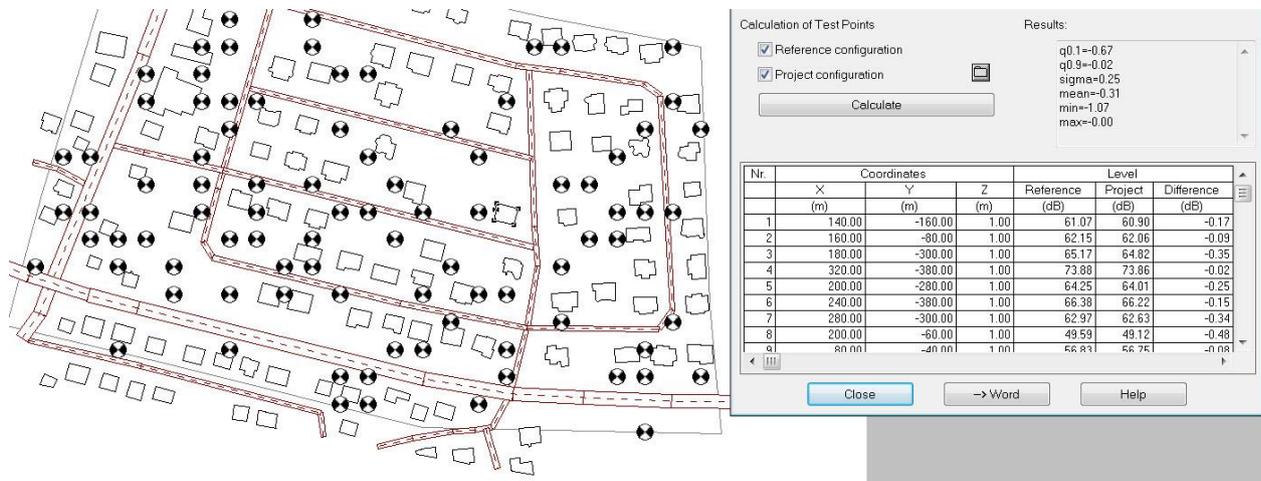


Figure 6: Receiver points distributed automatically according to standardized requirements and results of the statistical analysis

If this evaluation is done before the time consuming calculation of the noise map is started the benefits are obvious because it provides relevant information about the expected calculation time versus the deviation in the results.

4 Automated tasks applied to the presentation of results

As an application example the use of a Lua Script in order to export a noise mapping project to an online and interactive noise map (ONM) is explained. The ONM platform consists of two parts: the platform code – basically a HTML/JS/CSS website – acting as a template and the data of the noise scenario, which is exported from the project files. The platform code is executed within an end-user browser and therefore, a plain and simple HTML server is enough for hosting the ONM platform.

1.1 Exporting ONM data from the noise prediction software

The discussed example is a noise project called *project.cna* and consisting of four different scenarios with four evaluation parameters each. The last scenario contains the air pollution evaluation parameters NO_x, SO₂, PM-10 and Benzene which have been calculated in a grid with a different step size than the noise grids. Due to this fact there are discrepancies between the calculation points of both noise and air pollution maps.

The export of data is again performed by a Lua script. The objective here is the automation of the same operation on multiple files otherwise the operator would need to edit the original files several times and export manually each time. The main advantage of the script is that it can be adjusted to the individual needs of any project. As explained before in this project example several variants have been calculated for noise evaluation parameters with a grid spacing of 10m x 10m and the air pollution map has been calculated by using a 5 m x 5 m grid spacing.

The script allows the desired oversampling so all grids are adjusted to the right spacing. The structure of the script is divided into sections which are briefly explained below:

- Specification of the input and output paths.
- Definition of the variants. Within this section, the name of every variant that should be exported to the ONM platform is written. Each variant corresponds to one project file, including its corresponding grid. For example, the projectv1.cna is named as Scenario A.
- Check of the grid limits. The variants are previously loaded and a further check of the grid spacing is performed by writing out the limit coordinates.
- Exporting Routine. Finally variants are exported to Web-Bitmap format. This specific format creates a set of image tiles in different resolution steps which are coincident to the number of zoom steps that the ONM interface will have available. In this section an automatic readjustment of the grid spacing will be executed in case of grids with different spacing.

As a result of the export many information – map tiles exported as bitmap tiles with different resolutions - will be saved into the output path.

1.2 Data integration into the ONM platform template

The last step is the integration of data into the ONM platform. Since it is a template, the platform always has the folder structure shown in Figure 7:

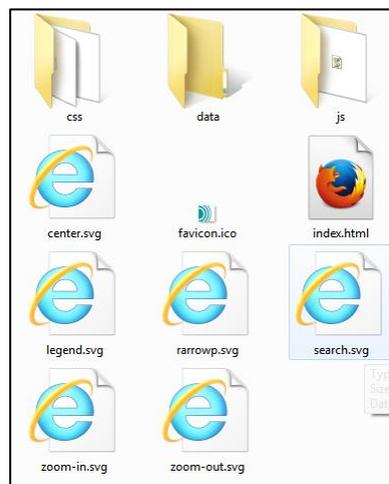


Figure 7: Folder structure of the ONM platform. Exported files need to be saved into the “data” folder

All exported files are simply and directly – without using sub folders - saved into the folder named “data”.

1.3 Description of the ONM platform interface

Starting up the ONM platform project only needs the allocation of the folder structure shown in Figure 7 onto a web server. Once the ONM has been started, the web interface is accessible from any web browser and offers different features like the display of the color legend and the navigation panel. In addition there are drop down menus to switch over the different variants and evaluation parameters – including those related to air pollution -. In all cases the calculated value is updated and displayed in a window close to the mouse cursor.



Figure 8: Online Noise Map web Interface. Noise Map L_{day} of Variant A



Figure 9: Air Pollution Map for NOx

The ONM web interface project also includes some features not present on the noise mapping software. The most relevant is the search address function to look for streets provided their name was included project file under the name field of the roads. This feature is in line with the requirements of the article 9 of the EU Noise Directive [2]:

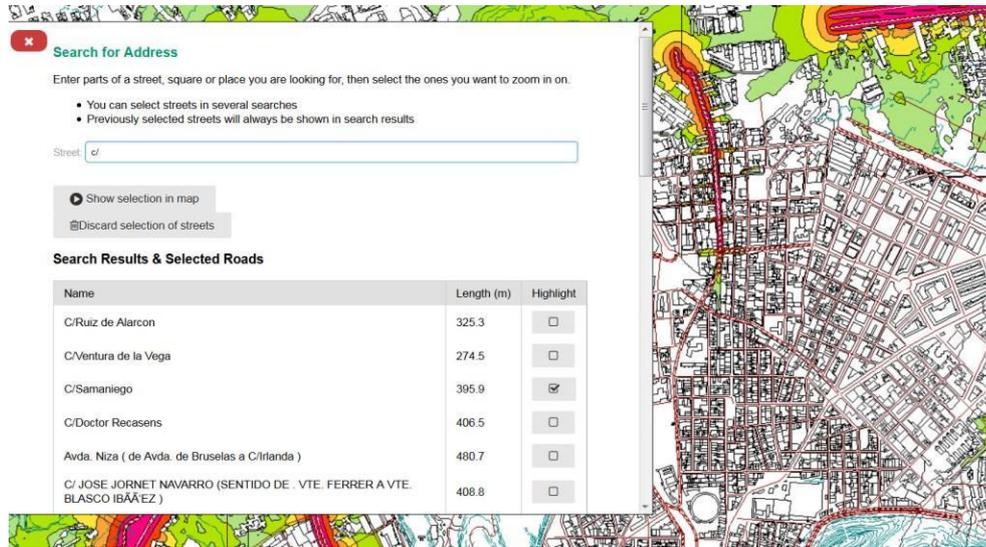


Figure 10: Search for Address dialog. The ONM shows on screen the search result of streets present in the model

5 Conclusions

The task automation through scripts applied to noise prediction software demonstrates to be a powerful tool providing additional features usually not present in noise prediction software user interfaces. This paper demonstrates a summary of benefits of these techniques applied to the different stages of a noise mapping project such as the import and conditioning of data, calculation or presentation to the public compared to the standard procedures normally used.

References

- [1] Commission Directive (EU) 2015/996, establishing common noise assessment methods according to Directive 2002/49/EC of the European Parliament and of the Council, 2015.
- [2] Directive 2002/49/EC of the European Parliament and of the Council, relating to the assessment and management of environmental noise, 2002.
- [3] OpenStreetMap Foundation (OSMF) [Online], supporting the Open Street Map Project. http://wiki.osmfoundation.org/wiki/Main_Page
- [4] DataKustik GmbH, CadnaA – Software for Environmental Noise: www.datakustik.com
- [5] Information about Lua language [Online.] available: <http://www.lua.org/docs.html>
- [6] Acoustics – Software products for the calculation of sound propagation outdoors – Quality requirements and test methods, German Standard DIN 45687