Version1, 22/04/2021

ViewShed Analysis Documentation

1. Overview:

The system is intended to calculate and present a line of sight (LOS) analysis from a set of pre-defined points (Antennas) to any point in a chosen geographical region (Transmitter). The results Depend on height of the antenna, the height of the transmitter and a Digital Elevation Map (DEM) file that represent the surface height in the region. This tool doesn't take into account the affect of earth curvature, this effect might be significant (unless you believe the earth is flat) and its neglect should be considered (a 10 m high antenna has its true horizon at about 11 km when no obstacles obscure it). The package includes several actions:

a) Calculation of a Viewshed layer, a Boolean raster layer for a specific antenna and a transmitter height, it set a TRUE (visible) or FALSE (invisible) value to all points in the region. This method includes a parallel calculation to increase the efficiency of the process.
b) Visualization of the cumulative viewshed. Using a set of precalculated raster layers and a choice of active antennae, the function sums the number of observant antennae at any point and present this value on an interactive map.
c) Visualization of intersecting line of sight. Using a set of precalculated raster layers, a choice of observant antennae and blind antennae, the function identifies the points that obey these visibility conditions and present these on an interactive map.
d) The function can allow the user to choose an antenna and point on the map and present the surface height under a line connecting these two points. It also presents the detailed DEM at the vicinity of the line. This analysis allows the investigation of the obstacles between any two points to explain the LOS results

2. Installation

To run the program, one must have an installed R program and RStudio IDE (interactive development environment), instructions for these steps can be found in https://rstudio-education.github.io/hopr/starting.html.

Using the application can be divided into two steps; **a)** Calculating viewshed layers, that requires a list of antennas location and height above ground, and some R programing capabilities and **b)** presenting**,** viewing, and analyzing the precalculated LOS layers, this step does not require any R programing capabilities.

It is recommended to save the entire application in a single directory (commonly named "Viewshed", containing a R project "Viewshed" and several subdirectories as follows:

- DEM_Files – a file containing the DTM files needed to compute the LOS analysis (needed only for the calculation step).
- ANTLOC - a directory containing a csv file with antenna coordinates and height above ground analysis (needed only for the calculation step).
- LOSData – a directory containing the layers produced in the calculation step. Using the application for the presentation of the results requires 5 files for each choice of transmitter height, coverage area, antenna set and DEM resolution. The directory also contains the icons for antenna (See.jpg)

- Functions – a directory containing the set of functions required to calculate and present the analysis
- Documentation – a directory containing this file 😊

3. Calculation of viewshed layers

a. Preparing the system and required data.

To initiate the calculation, one should open the script file “computeViewshed.R” (located in “functions” subdirectory). The following paragraph will detail the functions and procedures a user should invoke (code is accentuated `distinguish font`) to complete the calculation, these commands are already part of the script and the user needs only to change specific input parameters that will be highlighted in the following lines:

- `source('functions/Viewshed.R')`
  - All functions are in the “Viewshed.R” file and are attached by this command.
- `setEnv()`
  - This function source external libraries and define several constants, several warnings usually appear in this process.
  - If an error appears: Error in library("XXX") : there is no package called ‘XXX’, the user is required to install the following package: `install.packages("XXX")` (common in first operation of the code)
- `box <- data.frame(lat=c(32.38,32.68),lon=c(35.27, 35.68))`
  - The coverage map is calculated only within a predefined region, here the geographical coordinates of two diagonal corners of the region should be set.
  - Note that larger area requires larger computational effort and therefore longer calculation times.
- `DEM_name<-"DEM_Files/DEM_Harod.tif"`
  - If the user has the elevation data of the region (digital elevation map- DEM) on a geographic grid, he can specify it here, other wise the data can be freely download from the web by specifying `type="web"` in the `getDEM` command.
  - The DEM should be given in TIFF format, see *Obtaining DEM data* section for more details.

- `ANTfilename <- "ANTLOC/HarodANTSLoc.csv"`
  - Path to the antenna location csv file. The file should contain all antenna locations and must include the following variables:
    - "ANTName" – name in English letters, numbers and spaces are allowed.
    - "ID" – an integer number specifying the antenna.
    - "LAT", "LON" – geographical coordinates (wgs84 projection)
    - "Towerheight" – antenna height above ground in meters
- `cr <- getDEM(type="file",box,filename=DEM_name)`
  - Retrieve and cut DEM from the tiff file, in case the user wants to reduce the resolution, an argument `resoluton = XXX` where XXX is the required resolution in meters can be added to the `getDEM` function.

- In a case where the user doesn't have a high-resolution DEM file, the DEM can be obtained from the web: `cr <- getDEM(type="web",box)`, this usually provide a DEM with 3 arc resolution (~ 90 m)
- Note that the provided resolution has a dramatic effect on the calculation time and file size. Therefore, before dispatching the calculation with maximal resolution it is strongly recommended to begin with a low-resolution run.

- `ANTS.df <- setANTS(ANTfilename,cr)`
  - Reads and prepares the antenna details file. The file should contain all antenna locations and must include the following variables:
    - "ANTName" – name in English letters, numbers and spaces are allowed.
    - "ID" – an integer number specifying the antenna.
    - "LAT", "LON" – geographical coordinates (wgs84 projection).
    - "Towerheight" – antenna height above ground in meters
  - The function returns a SpatialPointsDataFrame object in the geographic, wgs projection, with the added variable "GroundASL" which is the ground altitude at the antenna position (in meters)
- `viewSetup(ANTS.df,cr)`
  - presenting the system setup: antennas, region map, and DEM. Click an antenna to see its details.

b. Invoking the calculation

After setting the required variables the system is ready for the calculation of the line-of-sight layers (LOSlayer) and saving it. As mentioned before, the provided resolution and region-size has a dramatic effect on the calculation time and file size. Therefore, before dispatching the calculation with maximal resolution it is strongly recommended to begin with a low-resolution, small region run.

The calculation can be launched either as serial calculation or parallel calculation (whenever a large computational power is available) the calculation run for a set of defined antennae and a set of transmitter heights and saves several required files for each transmitter height.

The serial code:

- `SerialComputeViewShed(layername,cr,ANTS.df,transAlts,ANTlist)`

The parallel code:

- `ParallelComputeViewShed(layername,cr,ANTS.df,transAlts,ANTlist)`

The input variables for both these functions are:

- `layername="XXX"`
  - A name for the files produced by this procedure (the same files will be used to present the LOS analysis). The function produces several files for each transmitter height (TransAlt0.3m as an automatic prefix for the 0.3 m high transmitter, for example):

- *ANTS_TransAlt0.3mXXX*
  - A csv file containing the details for the calculated antenna.
- *DEM_TransAlt0.3mXXX*
  - The DEM file used for the calculation (raster file).
- *LOSLayers_TransAlt0.3mXXX*
  - A stack of raster layers, each layer corresponds to a single antenna and contains TRUE / FALSE for each grid point according to the line of sight clearance.

- `ANTlist <- c("1","2","3"),`
  - A list of antenna ID's to run the calculation on, a user can specify `ANTlist <-as.character(ANTS.df$ID),` which will run the calculation on all antennas in the antenna csv file
- `transAlts <- c(0.3, 2, 5,10)`
  - list of transmitter heights to calculate, each height will be calculated for all antennas and saved as a single presentation set (5 different files).

it is recommended to run the following line before invoking the parallel calculation.

- `detectCores()`
  - detects how many cores are available on your machine.
- `registerDoParallel(3)`
  - assigns X processes for the following calculation.

c. Invoking the calculation

The calculation of a complete set of antennas might become demanding, especially at large region sizes and high DEM resolutions. Therefore the ability to edit a LOSlayers set is required, this ability includes the addition, removal as well as changing names (antenna name and ID) without recalculation of viewshed. substituting a single antenna:

- `AddRmAnts(final_name,base_str_name,add_str_name,ANTID2rm,ANTID2add)`
  - The function edit the `base_str_name` files, (note that you need only to provide the kernel name, without the "ANTS", "LOSLayers" or "DEM" prefix), it removes `ANTID2rm` antennas and adds `ANTID2add` from the file `add_str_name`
  - The function will not protect you from combining files that were calculated for different transmitter height (It will give you a warning if both strings begin with different name ("TransAlt").
  - The function cannot overwrite the previous files.
  -
- `listLOSFiles()`
  - List of viewshed files available in the LOSDATA directory, it trims the prefix (it shows only csv files, and doesn't check that indeed all required files are present "ANTS", "LOSLayers" or "DEM)

- `listAnts(str_name)`
  - List the calculated antennas in a set of LOS files for a given file-name
- A possible code would look like:

```
listLOSFiles()
str_name <- "TransAlt3m_Res900a1-6"
str_name_new <- "TransAlt0.3m_Res900a11-16"
str_name_out <- "TransAlt0.3m_Res900new-a11-16"
listAnts(str_name)
listAnts(str_name_new)

AddRmAnts(final_name=str_name_out,base_str_name=str_name,…
add_str_name=str_name_new,ANTID2rm = c(2,3),ANTID2add = c(15,12))
```

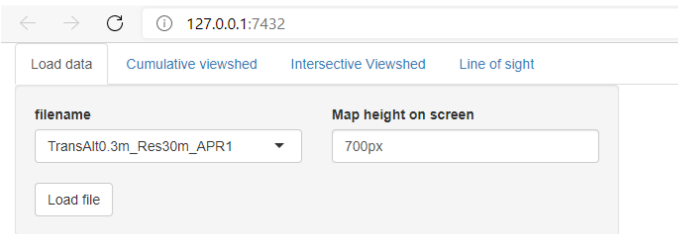
- `UpdateAntnames(final_name,base_str_name,initial_ANTID, newANTID,newANTname)`
  - Updates the names and ID's of antennas.
  - List of initial ID's, final ID's and new names should be provided (`newANTID`, and `newANTname` can also be omitted if only the ID or only the name should be changed).
  - Note no to give an ID that already exists!

4. Presentation

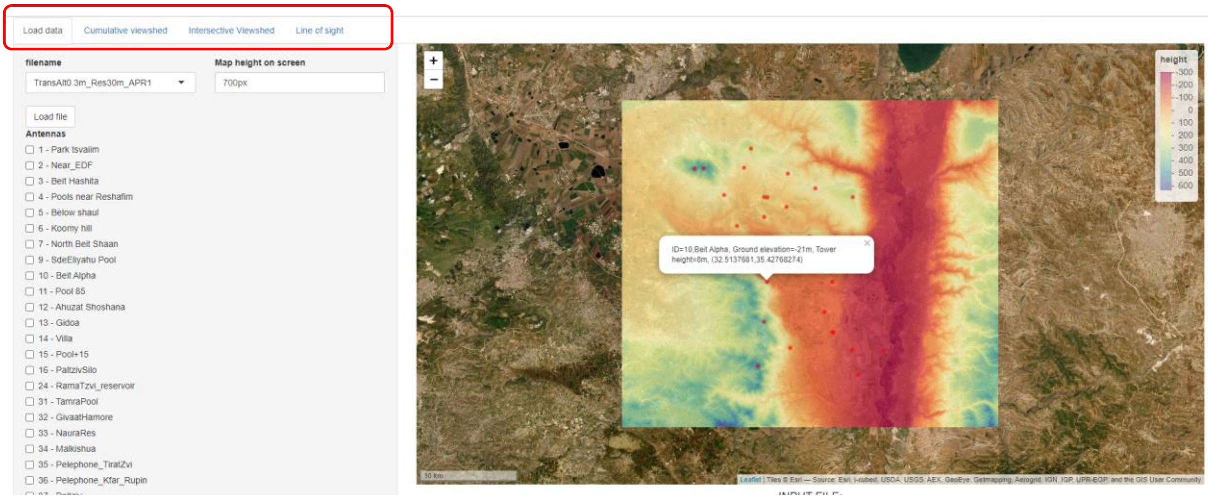
After the layers have been calculated, it is recommended to share the entire directory (including a single R project file) with the possible users of the application. Thus, any user with installed R program can use the application by simply executing the command `source('functions/ShinyViewshed.R')`, given that he has installed the required packages.

a. Loading a precalculated layer

Upon launching this command a web browser window will open:



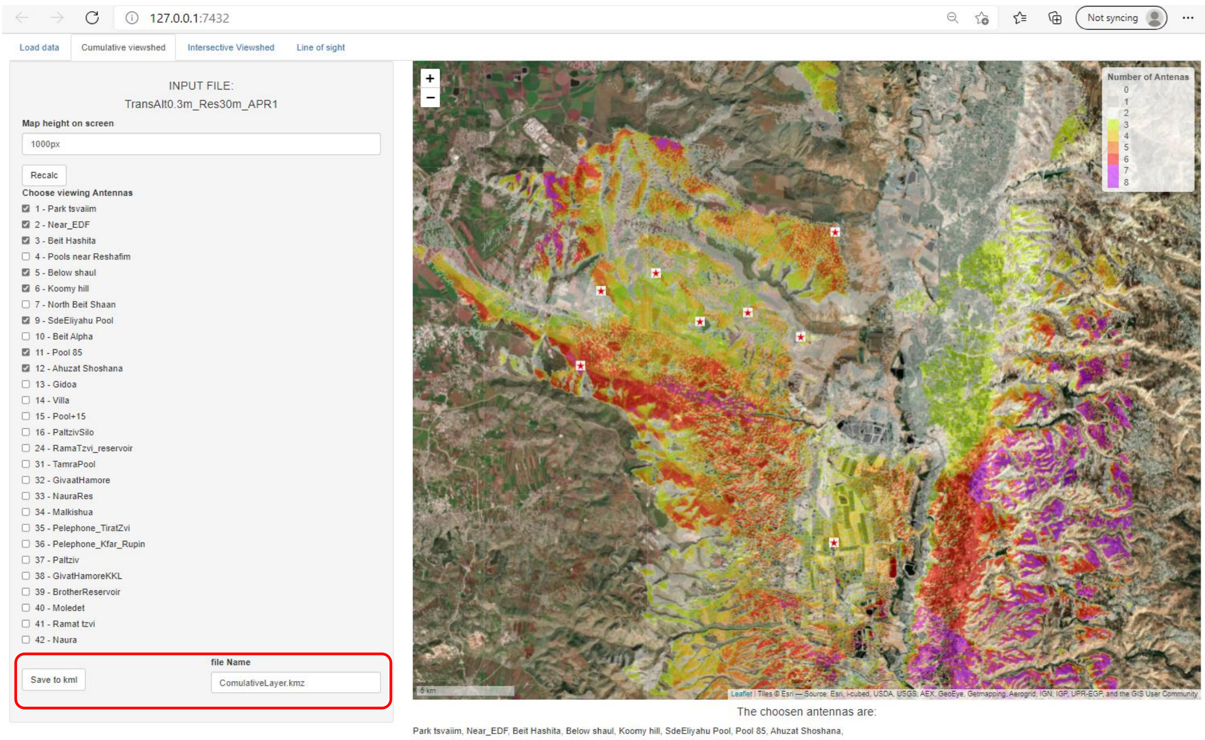
1. The user should choose a set of precalculated antenna file from the available list and hit "Load file"
2. The system will present the calculated region and antenna, clicking an antenna will present its details:



3. The “Map heigh on screen” input allows the user to present the map on the eniter screen for larger screens
   After verifying the basic data is as required, the user can navigate between the different tabs and present / analyze the data.

b. Cumulative Viewshed

Here the user can choose the antennae that contribute and present the number of visible antennae at any point in the region (this calculation is true for the pre-defined transmitter height)
The map will update when the “Recalc” button is clicked!
The participating antennea are marked as stars on the map.
The output can be exported in a kmz format to “file name” by clicking “Save to kml” at the bottom of the page!



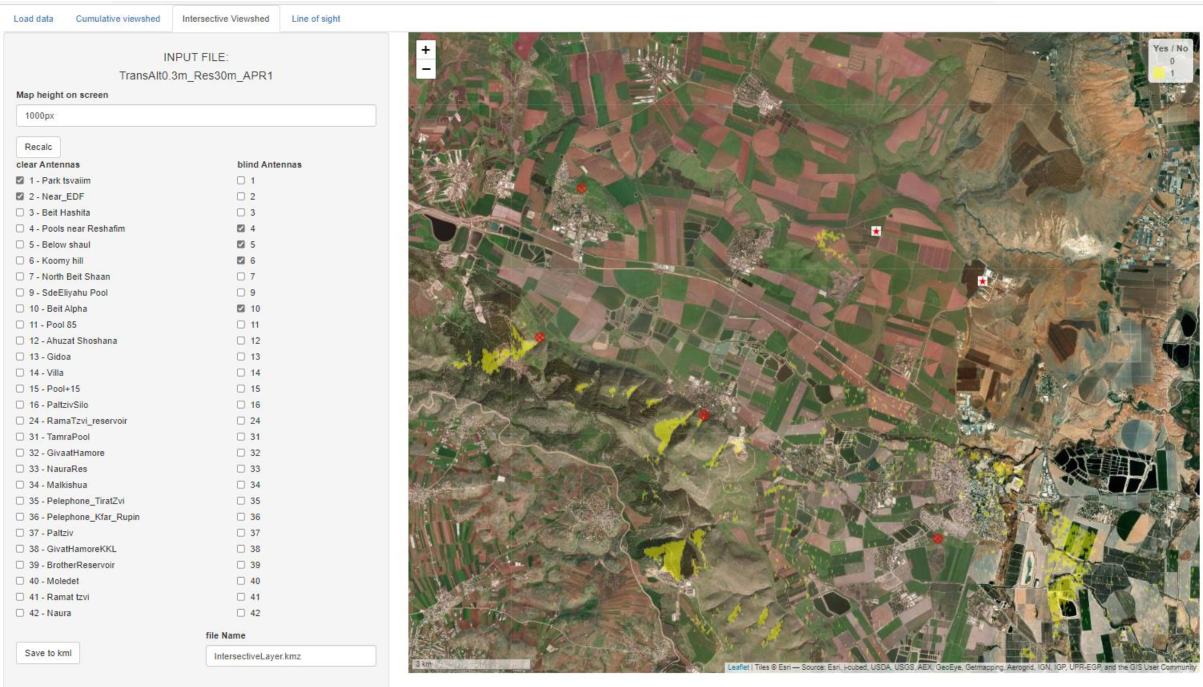
c. Intersective Viewshed

Here the user can present the intersection between **clear** (with LOS) and **blind** (without LOS) antennae. (This calculation is true for the pre-defined transmitter height)

The map will update when the “Recalc” button is clicked, area that agree with the defined condition will be colored yellow!

The clear antennae are marked as stars (★) on the map, while the blind are marked by circled cross (⊗)

The output can be exported in a kmz format to “file name” by clicking “Save to kml” at the bottom of the page!

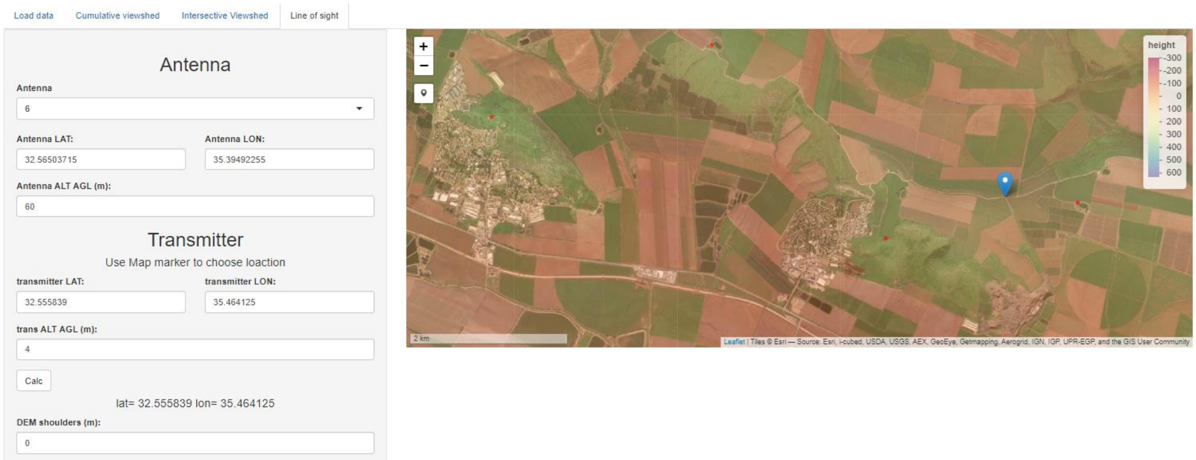


d. LOS analysis

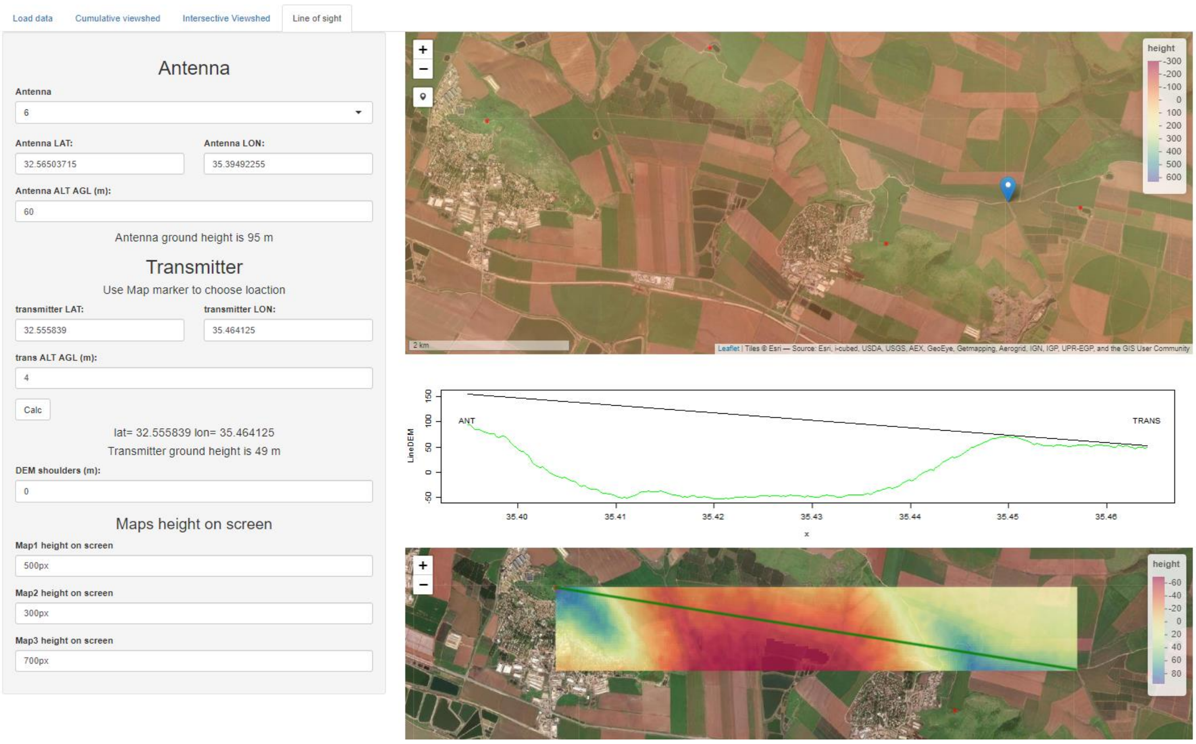
This tab allow some detailed analysis of the area and terrain. It plots the ground elevation under a liner between two points. It also present the DEM values in a square around this line (with user-chosen shoulders) , this option allows the high resolution inspection of the input DEM.

- Both points of the line can be inserted manually as a geographic wgs84 coordinates in the “LAT” and “LON” text input fields. Their elevation above ground can also be inserted manually.
- A second option for Antenna position input is to choose a predefined antenna from the underlying file.
- A second option for Transmitter position input is via the icon ( ) in the map. This icon will allow the user to point a location and will copy its coordinates and ground altitude to the corresponding fields.

- The “`DEM shoulders(m)`” value defines the margins around the DEM map



Upon clicking the “Calc” button, the line of sight between the two points will be displayed in a separate plot and a map that includes the zoom-in DEM around the line.



5. Obtaining DEM data

1 arc resolution (~ 30 meter) DEM files are available at http://sendimage.whu.edu.cn/res/DEM_share/SRTM1/N30,E30/ , a user can download his region of interest (as geographical anchored TIFF file) and then provide it to the `cr <- getDEM(type="file",box,filename=DEM_name)`

command. Some basic concatuating and saving commands might also be useful:

1. Read a TIFF file

```
imported_raster1 <- raster(DEM_name1)
```

2. Concatuate two DEM files

```
imported_raster <-
merge(imported_raster1,imported_raster2)
```

3. save a merged raster file locally for further use:

```
writeRaster(imported_raster, filename="DEM_Harod.tif",
format="GTiff")
```

6. **Dfasf**