Review of “HORAYZON v1.1: An efficient and flexible ray-tracing algorithm to compute horizon and sky view factor” by Steger et al

Main comments

The paper introduces the model HORAYZON for computating horizon lines and sky view factor (SVF) from digital elevation models which is relevant for land surface modeling/NWP. The model introduces a new algorithm based on the ray-tracing library `embree` to cope with demanding computational requirements of SVF computations in high-res modelling. Standard computations are notoriously slow since the SVF is a non-local quantity of the DEM.

The manuscript is well written, contains illustrative, high-quality figures, benchmark tests, comparisons to previous algorithms (Buzzi) and the application to three large DEM data-sets (NASADEM, swisstopo, USGS). Overall this paper is in good shape and ready for publication soon. I only have a few comments regarding the method and the code:

1. **SVF method.** I am a bit puzzled about the way the SVF is computed using the “modification” announced in line 247. Is this still exact or already an approximation? I do agree that the starting point (Eq 4) is the true/correct sky view factor, given that ϕ and ϑ are azimuth and polar angles (in standard spherical coordinates) in the *sloped* coordinate system where the surface normal is given by $\boldsymbol{n} = (0, 0, 1)$. But the “modification” (in the form of Eq 8,9) indicates that ϕ and ϑ in Eq 4 are already interpreted as the angles in the horizontal ENU coordinate system. While it is obvious that the present method refrains from simply calculating the SVF in the horizontal coordinate system (which is surely wrong) I cannot grasp if the method is a strict reformulation of the SVF in the sloped coordinate system. It appears to be in between. I think it is important to explicitly state if (and why) the present SVF formulation is an approximation or mathematically exact. This is linked to the statement l.271/172 where it is concluded that both computations give the same (even for the red-arrow point in Fig 6?). If the SVF is an approximation, this statement indicates how good the approximation is for the considered examples. If it is exact, this statement confirms the correctness of the implementation. These are two very different conclusions.

2. **Installation.** For me (using anaconda on linux) the installation required quite some trial and error to resolve version conflicts, in particular with the installation of GDAL. I highly recommend to improve the installation process by automatically installing packages in the correct version alongside via `setup`.

3. **Examples.** When I wanted to run an example, I followed the README, downloaded data from swisstopo, adapted the `gridded_SwissALTI3D_Alps.py` and ran into the following error:

```
(horayzon) $ python gridded_SwissALTI3D_Alps.py
Warning: no tile found for e2683n1152
Tiles imported: 1 of 5476
Warning: no tile found for e2684n1152

[... many output lines of the same type ...]

Warning: no tile found for e2756n1225
Tiles imported: 5476 of 5476
Warning: Nan-values (-9999.0) detected
[28000.  7000.  3000.  7000. 28000.]
Size of quad domain: (8501, 8501), vertices: 0.87 GB
Size of full domain: (36501, 36501), vertices: 15.99 GB
Range (min, max) of (scaled) DEM data: -159984.0, -159984.0 m

Traceback (most recent call last):
```

```
  File "/home/loewe/devel/python/horayzon/HORAYZON-main/examples/gridded_SwissAl
    raise ValueError("(Scaled) DEM range too large -> issue for uint16 "
ValueError: (Scaled) DEM range too large -> issue for uint16 conversion
(horayzon) $
```

Unfortunately I didnt have the time to debug this any further, I guess it is just a problem of missing input. (How much data do I have to download in fact to get the example running?) Anyway, from my experience, the best motivation for a future user to work with a model is to provide a plug-and-play example. Here a generic DEM (not subject to license restrictions) in the correct input format could be supplied together with the code for getting started, e.g. the crater DEM. I recommend to improve the user-friendliness of the example, this will greatly support use of the model (which has a catchy name, btw).

Kind regards,
Henning Löwe

Other comments

(l46): This is a bit misleading (?) As far as I understand, also the present method also works with gridded data.

(l315): Figure 7 should contain the extrapolated curves from Buzzi up to 10^6 grid cells for following this statement here.

(l321): This is not obvious from Fig 7. The speedup seems to approach a constant asymptotically.

(l353): How exactly is the gridded data converted to a triangular surface mesh in the ray casting? If this step requires interpolation to obtain a closed surface it should be stated.

(l365): r $\rightarrow r$

(Fig 10/11): Given the elevation map in (a), I dont understand the occurrence of several "white" spots in valley bottoms where SVF= 1. In particular the one in the lower left corner of Fig 11 (d). This location is surrounded by quite a pronounced crest line for almost the full azimuth range. So how can this lead to a sky view equal to unity?