The Journal of Open Source Software

SOLWEIG-GPU: GPU-Accelerated Thermal Comfort Modeling Framework for Urban Digital Twins

Harsh G. Kamath [1*¶], **Naveen Sudharsan** [1*¶], **Manmeet Singh** [1], **Nils Wallenberg** [2], **Fredrik Lindberg** [2], **and Dev Niyogi** [1,3¶]

1 Jackson School of Geosciences, The University of Texas at Austin, USA **2** Department of Earth Sciences, University of Gothenburg, Sweden **3** Cockrell School of Engineering, The University of Texas at Austin, USA ¶ Corresponding author * These authors contributed equally.

DOI: 10.21105/joss.09535

Software

- Review
- Repository
- Archive

Editor: Adithi R Upadhya

Reviewers:

- @lukasbeuster
- @ruedi99ms

Submitted: 04 September 2025
Published: 13 February 2026

Summary

Several tools exist for modeling outdoor thermal comfort and radiative environments in cities. For example, RayMan performs computations at the point scale using simplified representations of urban geometry and is computationally efficient; however, it does not provide spatially continuous fields. ENVI-met is a three-dimensional prognostic model that resolves spatially explicit urban microclimate processes, but it is computationally expensive and has a closed-source implementation. In this context, we present Solar and LongWave Environmental Irradiance Geometry–Graphics Processing Unit (SOLWEIG-GPU), a Python package that enables execution of SOLWEIG version 2022a on GPUs via a single-line code interface, command-line interface (CLI), or graphical user interface (GUI). The package enables GPU-parallelized human thermal comfort modeling at meter-scale resolution across city-scale domains by computing key variables, including sky-view factor (SVF), mean radiant temperature (TMRT), ground shading, and the universal thermal climate index (UTCI). SOLWEIG-GPU is intended for use by academicians and researchers, as well as city weather, sustainability, and heat-risk practitioners. Sample data are available on Zenodo, and a quick-start guide is provided in the documentation.

Statement of need

The original SOLWEIG model (Lindberg et al., 2008) was developed to calculate TMRT over small geographical areas in cities. At a small spatial scale, based on the time required for computation, the model could be run on CPUs (Kamath et al., 2023). However, for city-scale thermal comfort estimation, the model can be accelerated using a GPU. Specifically, the calculation of the SVF (the most time-consuming step), TMRT, and UTCI can be processed on a GPU. Currently, there is a tool that computes SVF on GPU, but it uses Python to read the inputs and write the output rasters, while the SVF calculation on GPU is performed by interfacing with C code (Li & Wang, 2021). Thus, there was a need for a Python-based end-to-end framework to run SOLWEIG on a GPU. Our framework is implemented in PyTorch, which automatically selects the GPU when available.

Functionality

SOLWEIG-GPU requires the following inputs: (i) Building digital surface model (DSM) that includes both buildings and terrain height (m), (ii) Digital elevation model (DEM) that is the bare ground elevation (m), (iii). Tree or vegetation DSM that only represents the vegetation height (m) (iv). UMEP (Lindberg et al., 2018) style ground cover (optional), and

The Journal of Open Source Software

(v) meteorological forcing. Allowed land cover types are asphalt or paved, buildings, grass, bare soil, and water. However, users can add their own land cover types, provided they are familiar with the radiative properties of the surfaces (e.g., albedo and emissivity). All input datasets must have the same spatial resolution, projection, and spatial extent. The recommended projection is the Universal Transverse Mercator (UTM). For example, EPSG: 32614 (for Austin, TX). Necessary meteorological variables for SOLWEIG are: (i) 2-meter air temperature (*℃*), (ii) relative humidity (*%*), (iii) barometric pressure (*kPa*), (iv) downwelling shortwave radiation (*W/m2*). Additionally, near-surface wind speed (*m/s*) is required for UTCI computation. Longwave radiation is estimated using 2-meter air temperature, relative humidity, and pressure.

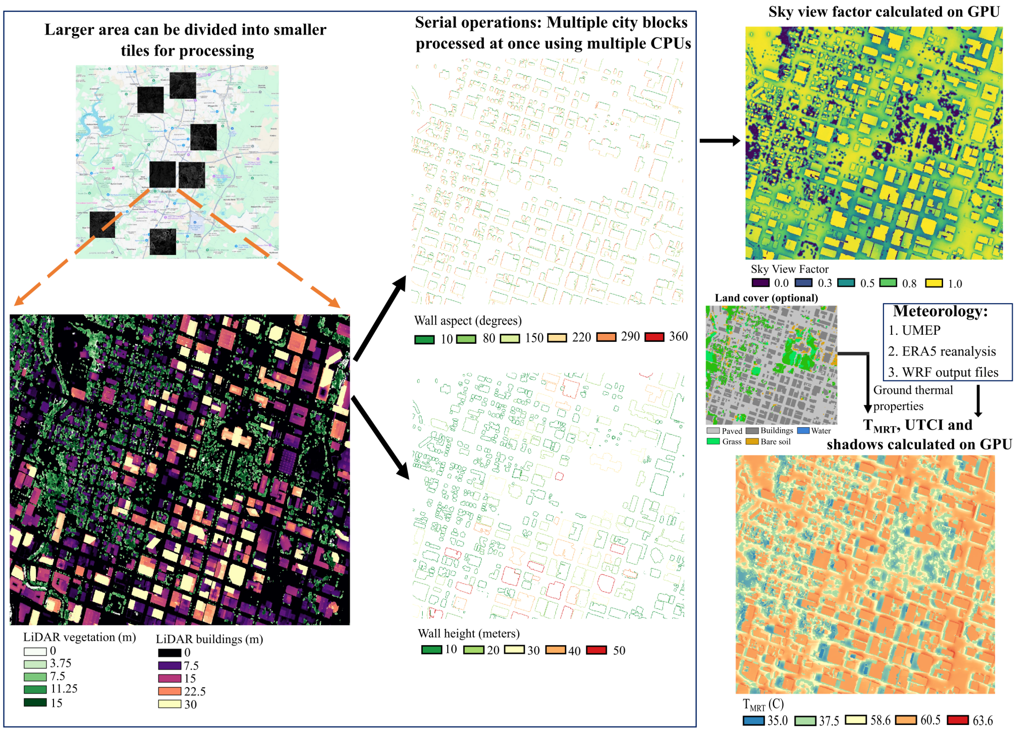

Figure 1: Different steps involved in calculation of `thermal comfort` using SOLWEIG-GPU: CPU and GPU-based calculations in SOLWEIG-GPU are shown.

Figure 1 shows the workflow of SOLWEIG-GPU, and a detailed description of functionalities is as follows: 1. SOLWEIG-GPU can divide a larger geographical domain into smaller tiles and create separate meteorological forcing for each of the tiles. 2. If **torch.cuda.is_available()** returns **'True'**, the simulations will be performed on a GPU. 3. The calculations of wall height and aspect (wall directional orientation) are faster on the CPU as they use an *i* and *j* indices to loop through the input rasters. Thus, we have parallelized this operation, with calculations performed across multiple CPUs. 4. SVF, TMRT, UTCI, and ground shading are computed on the GPU, if available. Additionally, the model can output shortwave and longwave radiations in both upward and downward directions. Users can select the required output variables from SOLWEIG-GPU, but UTCI is outputted by default. Note that the calculation of UTCI utilizes wind speed from the meteorological forcing; however, more advanced methods for calculating wind speeds in urban areas are available (Bernard et al., 2023). On clear, hot days, TMRT is a dominant driver, yet wind can still change UTCI by several ℃, so relying on grid-averaged wind may not always have a minimal impact. 5. Ground cover classes can be optionally provided, and they are used to set up the grid for the surface thermal properties (Lindberg et al., 2016). 6. SOLWEIG-GPU can only work with hourly meteorological data at present. However, the simulation's time period is based on the meteorological forcing data provided. The model can accept the meteorological forcing in three ways: (i) Output from the MetProcessor tool in

UMEP (Lindberg et al., 2018), (ii) Gridded ERA-5 reanalysis, and (iii) Gridded output from the Weather Research and Forecasting (WRF) model.

Usage

SOLWEIG-GPU can be run using Python code, CLI, or GUI, depending on the code adaptation. Details on model usage can be found in the SOLWEIG GPU documentation (https://solweig-gpu.readthedocs.io/en/latest/?badge=latest)

Documentation on outputs and visualization: https://solweig-gpu.readthedocs.io/en/latest/outputs.html

Examples: https://solweig-gpu.readthedocs.io/en/latest/examples.html

Comparison with SOLWEIG - CPU

UTCI simulations were run with different tile sizes, both on the CPU and on the GPU. The CPU-based simulations were run on a Windows 11 machine with a 10th-generation Intel Core i7 (i7-10700) and 16 GB of Random-Access Memory (RAM). The GPU-based simulations were run on an Ubuntu machine with an NVIDIA A6000 GPU (48 GB vRAM). Table 1 below reports the average time required for UTCI calculations (mean across 4-5 simulations with the same tile size). Note that Table 1 reports only the time for SVF and UTCI calculations, as the calculations for wall height and aspect are CPU-based.

Table 1. Comparison of time taken for UTCI computation using CPU and GPU-based machines for different tile sizes.

Tile size	CPU-based	GPU-based	GPU acceleration
1000 × 1000	1187 seconds (0.33 hrs)	47 seconds	~25×
1500 × 1500	3322 seconds (0.92 hrs)	105 seconds	~32×
2000 × 2000	6487 seconds (1.8 hrs)	158 seconds	~41×

Acknowledgments

This research is supported by NOAA NIHHIS [NA21OAR4310146], NSF AUDT [2324744], NASA Austin CARES [80NSSC25K7417], NIST [60NANB24D235], NASA POWERAI [80NSSC26K0190], UT UNESCO Chair and International Initiative [Sam Kumar Family and Ram Bashyam Family Gift]. The authors acknowledge the use of the Texas Advanced Computing Center under the project allocation ATM23014.

References

Bernard, J., Lindberg, F., & Oswald, S. (2023). URock 2023a: An open-source GIS-based wind model for complex urban settings. *Geoscientific Model Development*, *16*(20), 5703–5727. https://doi.org/10.5194/gmd-16-5703-2023

Kamath, H. G., Martilli, A., Singh, M., Brooks, T., Lanza, K., Bixler, R. P., Coudert, M., Yang, Z.-L., & Niyogi, D. (2023). Human heat health index (H3I) for holistic assessment of heat hazard and mitigation strategies beyond urban heat islands. *Urban Climate*, *52*, 101675. https://doi.org/10.1016/j.uclim.2023.101675

Li, X., & Wang, G. (2021). GPU parallel computing for mapping urban outdoor heat exposure. *Theoretical and Applied Climatology*, *145*(3), 1101–1111. https://doi.org/10.1007/s00704-021-03692-z

Lindberg, F., Grimmond, C., Gabey, A., Huang, B., Kent, C. W., Sun, T., Theeuwes, N. E., J"arvi, L., Ward, H. C., Capel-Timms, I., & others. (2018). Urban multi-scale environmental predictor (UMEP): An integrated tool for city-based climate services. *Environmental Modelling & Software*, *99*, 70–87. https://doi.org/10.1016/j.envsoft.2017.09.020

Lindberg, F., Holmer, B., & Thorsson, S. (2008). SOLWEIG 1.0–modelling spatial variations of 3D radiant fluxes and mean radiant temperature in complex urban settings. *International Journal of Biometeorology*, *52*(7), 697–713. https://doi.org/10.1007/s00484-008-0162-7

Lindberg, F., Onomura, S., & Grimmond, C. (2016). Influence of ground surface characteristics on the mean radiant temperature in urban areas. *International Journal of Biometeorology*, *60*(9), 1439–1452. https://doi.org/10.1007/s00484-016-1135-x