Simple and low-cost horizon estimation method for solar harvesting systems

Kha Boa Khanh CAO, Vincent Boitier

Simple and low-cost horizon estimation method for solar harvesting systems

Kha Bao Khanh CAO, Vincent BOITIER
LAAS, University of Toulouse
kha-bao-khan.cao@laas.fr, vboitier@laas.fr

***Abstract*—Sky imaging is used to determine local obstructions that may have an important impact on solar availability of an autonomous system powered by photovoltaic (PV) energy. However, there is currently no inexpensive, yet simple, solution that incorporates the determined horizon with weather database and system specifications for forecasting purposes. In this work, we first look at the autocalibration of cheap fisheye lens clipped onto a smartphone to extract a complex horizon from a sky image. It is then coupled with PVGIS irradiance database to generate a forecast. Finally, we contrast this result with experimental ones taken during March 2023. Overall, the shading information improves forecast precision but error still persists due to factors like weather variability and the difference between irradiance received by the solar panel's large surface and what is captured by a tiny sensor.**

***Index Terms*—PV forecast, horizon estimation, fisheye sky imaging**

I. Introduction

Solar energy forecast is essential to ensure the continuous operation of an autonomous system powered by PV energy. A short recap of elements impacting their performance can be found in Figure 1, among which the effect of nearby obstructions is the scope of this work. The objective is to evaluate how much shadow is cast over the system with the constraint of low budget and simplicity. The target public is mainly non-experts who wishes to harvest solar energy like biologists seeking to power a bird camera in the forest where branches may cover parts of the solar panels.

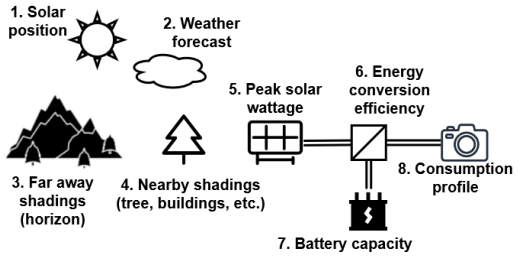

Fig. 1. Elements impacting solar energy available for autonomous PV systems

Three classes of shading estimation are widely applied in both industrial and academic contexts: 3D CAD, Digital Surface Models (DSM) and sky imaging. 3D CAD software are powerful but complex, more suitable for dedicated solar engineers than non-experts. As for DSM based methods, while they circumvent the need to scout out the physical location, depend on high quality geographical database that is not easy to come by. Therefore, sky imaging is the technique of choice given our constraints.

The idea is to take a photo of the sky and, by knowing how and where it was taken as well as the camera lens parameters, determine if and when there is obstruction to sunlight. A lot of commercial tools already exist [1], but so far no low-cost solution (i.e. < 100€ approximately) directly integrates the acquired horizon profile with weather databases and PV system specifications. In the literature, the usage of clipped-on fisheye lens for smartphones has proven to be successful by the authors of [2], but their fisheye parameter calibration steps are still relatively complex. Another research developed an application called Solar Survey that succeeded with the aforementioned integration, but the lack of continued development from the author has left the program unusable on newer Android devices [3].

We intend to propose improvements to reach a fisheye imaging solution that will be both easy to maintain, easy to use and accessible to everyone: the complete setup only involves a Python program, a smartphone with a clipped-on fisheye lens and a calibration pattern for camera's parameters' extraction (Figure 2). The first section of this paper discusses fisheye lens calibration using machine vision techniques, followed up with a short description of the program's overall architecture. Finally, we will contrast the forecast with experimental results taken during March 2023.

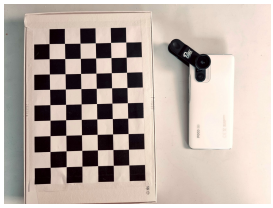

Fig. 2. Complete setup for shading estimation: a calibration pattern on the left, a smartphone with clipped-on fisheye on the right.

II. Fisheye lens autocalibration

The principal behind lens autocalibration is by having multiple photos of known world object points (i.e. vertices of a chessboard pattern shown in Figure 2), the algorithm can optimize the lens parameters to obtain the functions describing

the transformation of 3D world points to 2D image points. We have tested 3 different libraries: OpenCV Fisheye module [4], OpenCV Contrib Omnidir module [5] and Python port of Scaramuzza's toolbox [6] [7], with the last one having lowest remapping error. As can be seen in Figure 3, there is no noticeable error between the real world elevation denoted by our homemade protractor and the estimated elevation in the approximate 150° field of view.

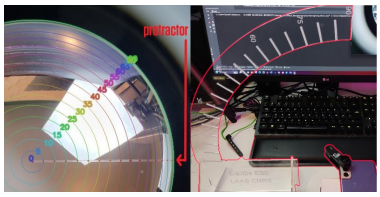

Fig. 3. Right: Photo of a protractor with red dots denominating real elevation overlaid with elevation estimated by the program. Left: overview of the test setup.

III. Incorporating local shadings with PVGIS

PVGIS [8] is a free and powerful tool that provides us with forecast on the performance of PV systems with relatively good precision [9]. It, however, has a limitation in how it interprets horizon data. The tool receives an array of points where each one denotes the terrain height at a specific direction. Therefore, it cannot describe objects like trees that may cover large portion of the sky above the solar panels but leaves direct sunlight access at lower elevations.

To ensure that all type of shades are handled, we utilize AstroPy [10] to calculate the hourly position of the sun and via the image, determine whether the sun is obstructed or not. For this, the original picture first has to be converted to black and white, where a white pixel showing an opening toward the sky and a black pixel representing an obstruction (Figure 4). To compensate for diffuse irradiance, we took the proportion of the clear sky over the total field of view. As for the direct irradiance component, we create a mask of pixels representing the solar path taken during an hour and calculate the percentage of black pixels present. This information is then incorporated to the irradiance data fetched from PVGIS using a Python API function.

Fig. 4. Sky image above our solar panel test setup. Original version on the left, converted to white and black to distinguish sky and obstructions respectively on the right.

IV. Preliminary results March 2023

We conducted a test by measuring the irradiance received by a 20W solar panel placed horizontally to the ground at the location as seen in Figure 4, oriented to 138° (South-East) and inclined by 8°. The test window was from 22 February to 26 March and we obtained an estimated 1173Wh of harvested energy in this period. Raw irradiance data was taken as an average of the same period between 2008 and 2016 in PVGIS. It expects an average 2252Wh of energy (91% error) for the same period. Compensated for obstructions, the estimation drops down to 1405Wh (19% error). Overall, this is an improvement, but the error is still relatively important which warrants a closer look at the hourly data.

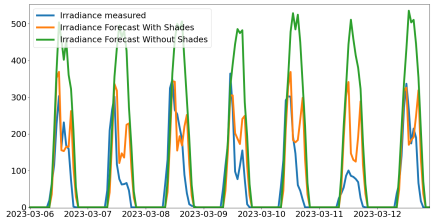

Fig. 5. Experimental results and forecast result (with and without shadings considered) from 06 March to 12 March

Observing the forecast in the period from 06 March to 12 March (Figure 5), we see that there is an obstruction between 13h and 15h, and its effect is also clearly visible in the measured irradiance on most days except for 11 March. This phenomenon is recurrent in our data set, meaning that overall, while the impact of shading was well taken into account, the test window could not filter out the effect of outlying weather conditions. Furthermore, the image shows shadings to the small camera sensor that may not accurately reflect what happens to the solar panel as a whole.

V. Conclusion & Perspective

The method showed significant improvements to the forecast precision (error reduced by a factor of 4.7) while staying very low-cost and simple, but extra verification is needed like comparing estimated and captured solar position on the image, and comparing our experimental data with a "forecast" based on irradiance taken in March 2023.

References

[1] S. Duluk, A. Kwok, and N. Heather, "Comparison of Solar Evaluation Tools: From Learning to Practice," 2021.
[2] M. J. Oliveira Panão, R. F. Carreira, and M. C. Brito, "Determining the shading correction factor using a smartphone camera with a fisheye lens," *Solar Energy*, vol. 190, pp. 596–607, Sept. 2019.
[3] J. A. Ranalli, "Solar Survey: Development and validation of a smartphone-based solar site assessment tool," *Solar Energy*, vol. 122, pp. 1199–1213, Dec. 2015.
[4] OpenCV, "OpenCV: Custom Calibration Pattern for 3D reconstruction."
[5] OpenCV, "OpenCV: Fisheye camera model."
[6] T. Pönitz, "pyomnicalib."
[7] Scaramuzza, "Davide Scaramuzza - OCamCalib: Omnidirectional Camera Calibration Toolbox for Matlab."
[8] J. R. C. EU, "JRC Photovoltaic Geographical Information System (PVGIS) - European Commission."
[9] D. D. Milosavljević, T. S. Kevkić, and S. J. Jovanović, "Review and validation of photovoltaic solar simulation tools/software based on case study:," *Open Physics*, vol. 20, pp. 431–451, Jan. 2022. Publisher: De Gruyter Open Access.
[10] Astropy, "Astropy."