EMERGING TECH CONFERENCE – Edge Intelligence

Volume 01, 2022, Page 161 – 168

Proceedings of Emerging Tech Conference: Edge Intelligence 2022

Towards Solar Irradiance Forecasting Using Deep Learning on the Edge

E. A. Papatheofanous[a], V. Kalekis[a], G. Venitourakis[a], P. Tz. Amrou[b], G. Konstantoulakis[b], P. Golemis[b], D. Reisis[a]*

[a]*Electronics Lab, Physics Dpt, National & Kapodistrian University of Athens*

[b]*InAccess Networks, 12 Sorou Str, 15125, Athens*

dreisis@phys.uoa.gr

Abstract

The appearance of clouds in the sky and their movement can cause significant differences in the sun's direct irradiance on the surface of the earth and significantly affect the energy production of photovoltaic (PV) parks. Irradiance forecasting can prevent perturbations on the power supply and the grid balance. This process utilizes a module that has as its input a sky image and estimates the irradiance with image processing algorithms based on neural network techniques. Aiming at a performance effective irradiance estimation module that is also efficient with respect to the hardware requirements and therefore suitable for the edge computing paradigm, the current paper presents the study towards the development of a Convolutional Neural Network (CNN) based model that is able to process a sky image and produce a single continuous value that represents the estimated irradiance for the particular sky image. The real-time performance is validated by the implementation on the edge-oriented Xilinx Zynq UltraScale+ MPSoC FPGA.

1. Introduction

Solar photovoltaic (PV) power stations have currently an increased fraction of the power supply produced by renewable energy systems. In the PV parks, knowing in advance the intensity of the incident solar irradiance is crucial for the control of the power production. This is especially the case when cloud effects will reduce the irradiance significantly so that the park has to activate batteries for these times or notify the grid control regarding the possibility of variations in power production. An attractive approach for addressing the above challenge in future installations is to include systems that will forecast the irradiance. Such a system will forecast the irradiance if it has as input the corresponding cloud forecasting and therefore, the module that estimates the irradiance out of a sky image plays a crucial role in the PV parks operation.

The above modules utilize image processing techniques that nowadays can grant accurate results regarding modelling the state of the clouds in the sky. The images used are high resolution frames produced by cameras with fisheye lenses (sky imagers) that are positioned at the PV park. Most of these systems take advantage of the ongoing evolution of the Machine Learning (ML) techniques and especially the Convolutional Neural Network (CNN) models. Researchers and engineers complete the training of a CNN by providing as input the sequences of images in past times along with the corresponding values of the irradiance of those times. Moreover, they focus on the improvement of the complexity of the CNN computations targeting edge computing implementations with real-time performance.

Aiming at improving the performance of the irradiance estimation process, the current paper presents the study towards the design of an image processing module for this purpose. The proposed module is a CNN-based model that accepts as its input a sky image and outputs a single continuous value that represents the estimated irradiance for the particular sky image. This proposed module can be extended to perform irradiance forecasting by assuming an additional external frame prediction component that can provide it with a future sky image. In this study we evaluate the performance of popular state-of-the-art CNN models on the Folsom, CA dataset [1], a publicly available dataset for PV applications. To demonstrate the real-time execution times and the employment of the solution towards a smart PV park with edge computing capabilities, we provide an implementation of the models of our study on an edge device, the Xilinx Zynq UltraScale+ MPSoC FPGA.

A relatively wide range of diverse sky image processing approaches towards irradiance estimation and forecasting for PV applications are proposed in the literature [2-5]. Works that focus on image regression for producing irradiance values using deep learning techniques are the ones that are most closely related to this paper. The work in [6] reports a comparative study of various deep learning model types that perform short-term irradiance forecasting using sky images. The authors in [7] utilize the well-known ResNet model to perform image regression for irradiance estimation and forecasting processes. In [8] the authors propose a CNN-based image regression model that produces irradiance values for potentially replacing costly irradiance measurement equipment such as pyranometers. The work in [9] studies the performance of CNN and LSTM models for the task of image regression for irradiance mapping.

2. Dataset & Problem Description

In recent years, publicly available datasets have given a boost in the field of deep learning approaches towards irradiance forecasting. In the current work, we utilize the Folsom, CA dataset introduced in [1] and available in the repository [9]. The dataset covers the years 2014, 2015 and 2016. It consists of 1536x1536 images taken every minute from a 180∘ field-of-view sky imager located at Folsom, CA, USA. Along with the images, the dataset includes the irradiance measurements with the Global Horizontal Irradiance (GHI) being of particular interest. After performing the data cleaning, we arrange the image and irradiance pairs on a daily basis. The daily irradiance plot of a single day is shown in Fig. 1.

The daily irradiance follows an increasing trend from sunrise to noon, while it decreases down to zero in the evening following a pattern similar to that of the sun's elevation angle during the day. We observe that the smooth curve of the daily irradiance in this particular day is disrupted by cloud effects in the morning until noon. In Fig. 1 we mark the irradiance measurements that correspond to two significant dips and two clear sky moments and present the corresponding images in Fig. 2. In both dips the corresponding images show the effect of the sun being covered by clouds.

Apart from the computer vision applications where CNNs commonly perform image classification, CNNs can also handle the image regression task where they produce a single output with a continuous range of values instead of a classification result. The process of irradiance estimation from sky images is such an image regression task. In the current paper we study the performance of three popular CNNs on this process. For this, we train and evaluate the CNN models on the aforementioned dataset that consists of sequences of sky images and corresponding irradiance values. In this problem, the CNNs can potentially model sky image features such as the position of the sun in the sky, the appearance of clouds as well as effects of the sun being covered by clouds to produce an estimated irradiance value. The aforementioned irradiance estimation process is one step towards irradiance forecasting. Assuming an additional external module that can produce a future sky image, this image can be given as an input to the irradiance estimation CNN, which in turn will produce an irradiance value that will now constitute a forecast.

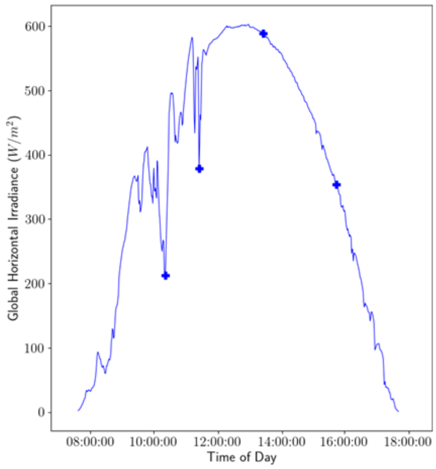

Figure 1: Daily irradiance plot for the day 3/11/2014 in UTC-7 timezone with markers on four particular measurements of interest

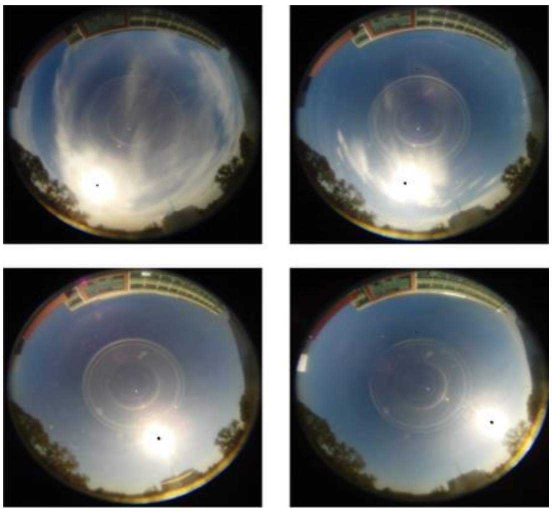

Figure 2: Sky images that correspond to the markers on the daily irradiance plot from left to right and top to bottom.

3. Implementation Flow on Edge FPGA

In this work, we aim to demonstrate the concept of a smart PV park with edge computing capabilities and with real-time performance for the irradiance estimation process. For this purpose, we provide an implementation of the CNN models on the Xilinx Zynq UltraScale+ MPSoC

FPGA which is tailored for edge applications. For the implementation of the CNN models we use the Xilinx Vitis AI development environment [11]. Vitis AI consists of several tools and provides a development flow that the developer can adjust to his own needs for rapid deployment of CNN models on Xilinx FPGA platforms. The Vitis AI development flow and summary of tools that we use is illustrated in Fig. 3.

The first step for deploying a custom trained model is to develop the application for the quantization of the model based on the Vitis AI Quantizer. The quantizer accepts floating-point model descriptions from well-known libraries such as TensorFlow and PyTorch and converts them to the corresponding fixed-point ones since the hardware accelerator on the FPGA operates on 8-bit values. Following the quantization, the developer compiles the quantized model with the Vitis AI Compiler to a set of instructions that will be executed on the hardware accelerator on the FPGA. For the hardware accelerator, Vitis AI provides the Deep-learning Processor Unit (DPU) IP Core that is implemented on the Programmable Logic (PL) of the MPSoC FPGA. The DPU is a programmable computing engine that is designed to efficiently accelerate a wide range of operations that are used in CNN models which target mainly computer vision applications. In the development flow, we configure the DPU in order to optimize the resources utilization of the FPGA and enable computational features for increased throughput such as the employment of two processing cores. Finally, we develop the application that will be executed on the Processing Subsystem (PS) of the FPGA which is an ARM-based processor. The application utilizes the Vitis AI Runtime library that exposes to the developer an API for the control of the DPU accelerator and supports functionalities such as asynchronous submission and collection of jobs and multi-threading.

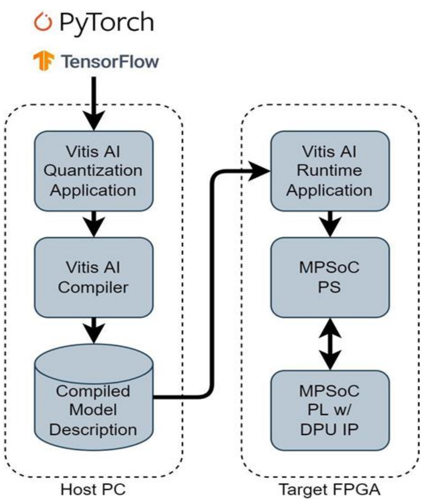

Figure 3: Vitis AI development flow and system illustrating processes performed both on the host PC as well as target FPGA.

4. Models Evaluation & Implementation Results

In the current work, we evaluate three popular CNN models on the process of irradiance estimation from sky images. The three models are the PyTorch provided implementations of ResNet-50 [12], MobileNetV2 [13] and SqueezeNet [14]. All these models include a final classifier part that consists of a single fully-connected layer with multiple output values that correspond to different classes. We note here that the irradiance estimation is not a classification task. Accordingly, we replace the aforementioned multiple output final fully-connected layer with a layer producing a single output value that corresponds to the estimated irradiance. The models are trained and evaluated on the Folsom, CA dataset described in Section 2. The training dataset consists of the years 2015, 2016 and includes 522320 samples of images with corresponding irradiance values while the test set consists of the year 2014 that includes 240944 samples. 20% of the training dataset is used for evaluation of the performance of the models during training. All the images of the dataset are resized to 128x128 pixels. The training procedure is the same for all the models. They are trained for 10 epochs with a batch size of 16 and with an adjustable learning rate starting at 10-4 and the Mean Square Error (MSE) validation loss. For each model, we perform evaluation during each epoch and select the checkpoint with the minimum validation loss to avoid overfitting. In Table 1, we summarize the results of the three models when evaluated on the test dataset of the year 2014, along with their size and number of multiply-add (MA) operations.

Models	RMSE (W/m^2)	nRMSE (%)	MAE (W/m^2)	# Params.	# OPs (MAs)
ResNet-50	64.92	15.79	38.73	23.51M	6.67G
MobileNetV2	81.43	19.81	50.16	2.22M	0.49G
SqueezeNet	70.31	17.11	44.68	0.74M	1.14G

Table 1. Evaluation of the three CNN models on the test dataset.

The metrics used for the evaluation of the performance of the models are the Root Mean Square Error (RMSE) and the Mean Absolute Error (MAE) which are calculated based on the irradiance values produced by the models and the ground-truth ones included in the dataset. The Normalized RMSE (nRMSE) metric is also calculated in order to provide an understanding of the proportional amount of error of the produced values compared to the ground truth. The model size as expressed by the number of parameters and number of operations is included in the table as well. We observe that the larger ResNet-50 model achieves the minimum RMSE and MAE with 23.51 million parameters and 6.67 GigaMAs. For this model, in Fig. 4 we present the results of the estimated and ground truth irradiance values for an entire day.

We observe that the model can identify the position of the sun in the image in order to produce a corresponding irradiance value following the pattern of the sun's elevation during the day. Furthermore, the model can also identify the effects of the sun being covered by clouds that result in dips in the measured irradiance. This indicates that the proposed approach can potentially forecast ramp events that heavily affect the PV power production.

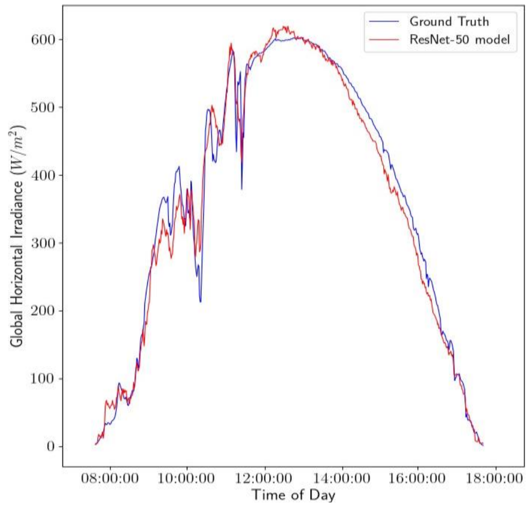

Figure 4: Daily irradiance plot for both ground truth and ResNet-50 model produced values.

Following the phase of the evaluation of the three different models, these models are implemented on the Xilinx ZCU104 FPGA board using the Vitis AI 2.0 development flow described in Section 3. The three models are benchmarked regarding their throughput in Frames per Second (FPS) when executed on the 2-core DPU. The results are shown in Fig. 5.

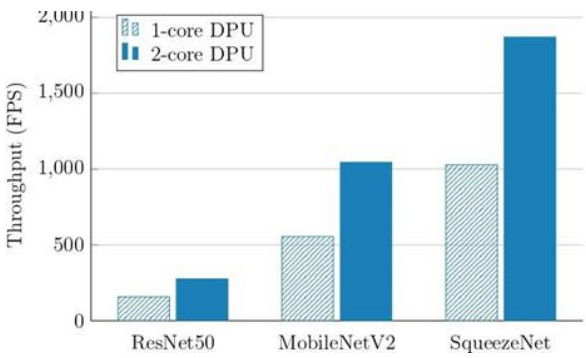

Figure 5: Throughput results for the three models of the study.

We observe that the larger ResNet-50 model showcases the lowest throughput at 158 FPS with a single core. The SqueezeNet model showcases the highest throughput at 1029 FPS even though its floating-point model performs more operations but with fewer parameters than MobileNetV2. By benchmarking all the models on both cores of the DPU using multithreading, their throughput roughly doubles showcasing the efficiency of the DPU. In all cases, such rates can be considered to meet the real-time requirement (e.g. of a camera providing a video at 60 FPS) leaving space for additional algorithms to complete the irradiance forecasting process.

5. Conclusions

In the current work we presented a study on the performance of well-known CNN models on the task of irradiance estimation from sky images as a step towards irradiance forecasting for PV power production applications. The results of 15.79% nRMSE of the ResNet-50 model on the test dataset shows the effectiveness of this approach as well as that there is space for improvement regarding the design of a CNN with increased irradiance estimation performance. Finally, we outlined the concept of an edge-enabled smart PV park by showcasing the real-time throughput capabilities of the CNN models when deployed on the edge-oriented ZCU104 board using the Vitis AI framework and development flow.

Acknowledgements

The research work of V. Kalekis, G. Venitourakis, P. Tz. Amrou, G. Konstantoulakis and D. Reisis was supported in part by the "Archon" project funded by the Greek GSRT programme ΕΣΠΑ, contract No Τ2ΕΔΚ-00864.

The research work of E. A. Papatheofanous was supported in part by the Hellenic Foundation for Research and Innovation (HFRI) under the 3rd Call for HFRI PhD Fellowships (Fellowship Number: 6220).

H.F.R.I. Hellenic Foundation for Research & Innovation

References

[1] Hugo T. C. Pedro, David P. Larson, and Carlos F. M. Coimbra. A comprehensive dataset for the accelerated development and benchmarking of solar forecasting methods, Journal of Renewable and Sustainable Energy 11, 036102 (2019)

[2] Yinghao Chu, Mengying Li, Carlos F.M. Coimbra. Sun-tracking imaging system for intra-hour DNI forecasts, Renewable Energy, Volume 96, Part A, 2016, Pages 792-799

[3] Niccolai, A.; Nespoli, A. Sun Position Identification in Sky Images for Nowcasting Application. Forecasting 2020, 2, 488-504.

[4] Kim, B.-Y., Cha, J. W., and Chang, K.-H. Twenty-four-hour cloud cover calculation using a ground-based imager with machine learning, Atmos. Meas. Tech., 14, 6695–6710, 2021.

[5] Rajagukguk, R.A.; Kamil, R.; Lee, H.-J. A Deep Learning Model to Forecast Solar Irradiance Using a Sky Camera. Appl. Sci. 2021, 11, 5049

[6] Quentin Paletta, Guillaume Arbod, Joan Lasenby. Benchmarking of deep learning irradiance forecasting models from sky images – An in-depth analysis. Solar Energy, Volume 224, 2021, Pages 855-867

[7] H. Wen et al. Deep Learning Based Multistep Solar Forecasting for PV Ramp-Rate Control Using Sky Images. IEEE Transactions on Industrial Informatics, vol. 17, no. 2, pp. 1397-1406, Feb. 2021

[8] H. Jiang, Y. Gu, Y. Xie, R. Yang and Y. Zhang, Solar Irradiance Capturing in Cloudy Sky Days–A Convolutional Neural Network Based Image Regression Approach. IEEE Access, vol. 8, pp. 22235-22248, 2020

[9] F. Wang et al. Deep Learning Based Irradiance Mapping Model for Solar PV Power Forecasting Using Sky Image. 2019 IEEE Industry Applications Society Annual Meeting, 2019, pp. 1-9

[10] Carreira, Pedro, Hugo, Larson, David, & Coimbra, Carlos. A comprehensive dataset for the accelerated development and benchmarking of solar forecasting methods. Dataset. Zenodo. 2019

[11] Xilinx. Vitis AI User Guide UG1414 (v2.5). https://docs.xilinx.com/r/en-US/ug1414-vitis-ai, 2022. Date accessed 15-10-2022.

[12] He, Kaiming and Zhang, Xiangyu and Ren, Shaoqing and Sun, Jian. Deep Residual Learning for Image Recognition. arXiv. 2015

[13] Sandler, Mark and Howard, Andrew and Zhu, Menglong and Zhmoginov, Andrey and Chen,

Liang-Chieh. MobileNetV2: Inverted Residuals and Linear Bottlenecks. arXiv. 2018
[14] Iandola, Forrest N. and Han, Song and Moskewicz, Matthew W. and Ashraf, Khalid and Dally, William J. and Keutzer, Kurt. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5MB model size. arXiv. 2016