G0401

Intraday solar irradiance forecasting using public cameras

Roy Sarkis (1), Ilker Oguz (1), Demetri Psaltis (1), Mario Paolone (1), Christophe Moser (1), Luisa Lambertini (1,2)
(1) EPFL, Lausanne/Switzerland
(2) Universita' della Svizzera Italiana (USI), Ticino/Switzerland;
roy.sarkis@epfl.ch

Abstract

With the significant increase in photovoltaic (PV) electricity generation, more attention has been given to PV power forecasting. Indeed, accurate forecasting allows power grid operators to better schedule and dispatch their assets, such as energy storage systems and reserve. In this paper, we propose a hybrid deep learning model and a convolutional neural network with memory, to provide intraday (2 hours) solar irradiance forecasts using sequentially-collected images from public webcams. The images are collected from two webcams available on EPFL's campus and capture a 360° view of the surroundings (sky and ground). The proposed neural network is trained on a year worth of data.

The performance of the proposed model is compared to those of a standard time-series forecast models, a linear regression as well as state-of-the-art neural networks trained on the same data set. We find that the proposed methodology outperforms all other models and matches the state-of-the-art methodology implemented by a global solar irradiance data modelling company while providing simplicity of implementation and efficient computation. We attribute this result to the information content present in the side view images of the sky and clouds unlike the top view of a satellite image. We expect that by adding more webcams and by combining information from satellite and weather models, we will further increase the solar irradiance prediction accuracy. This is attributed to our results showing that the use of two webcams instead of one decreases our root mean squared error.

The proposed methodology is very simple, computationally inexpensive and proves to perform very well to provide GHI forecasts for PV generation prediction. Improving GHI forecasting at the local level can be aggregated up to a regional or even national level and can help predict a country's imbalances in the transmission grid leading to optimal use of energy sources. With better intraday predictions, the dispatch plan (set a day in advance) could be followed more accurately and hence avoid imbalances that incur penalties and require use of non-renewable sources.

1. Introduction

In the last decade, the global usage of photovoltaics (PV) was the fastest developing renewable energy source with more than 1000 gigawatts of additional generation capacity installed [1]. This growth is mainly attributed to the decreasing costs of installations and smaller carbon footprint. However, the ongoing shift from traditional power sources in favor of renewables, like PV, requires the availability of an adequate and reliable capacity of regulating power. This challenge is addressed by achieving controllability over all the distribution networks so as not to deviate from their dispatch. This objective is achieved through the integration of decentralized generation and energy storage systems. This is evident in various frameworks including virtual power plants[1], active distribution networks[2] (ADNs) and grid-tied microgrids[3] which aim to provide local energy balance and ancillary services to the upper grid layer by using diverse resources [2]. Storage systems in ADNs have been used to offset the effect of stochastic and uncontrollable resources. This concept refers to the capability of the ADN to track a day-ahead power schedule, henceforth called *dispatch plan*. The optimal computation of the day-ahead dispatch plan and its intraday tracking, rely on the availability of PV forecasting tools along with the exploitation of energy storage systems operating at different time horizons [3].

As a matter of fact, the enhancement of ADN dispatch relies on the availability of high-performance forecasting tools of both electricity consumption and local power generation at both day-ahead and intraday time horizons along with storage systems of different energy reservoirs sizes and power regulation capabilities.

As known, the solar irradiance is quantified using the global horizontal irradiance (GHI) which represents the total solar radiation (including both the direct and diffuse radiation) incident on a horizontal surface. GHI is affected by many factors such as the time of the year, time of the day, weather, and, most importantly, cloud coverage.

GHI forecasting can be achieved using different techniques. The choice of the optimal method depends on the forecast horizon as well as the geographical scale [4]. These techniques are statistical methods, numerical weather predictions, sky images and satellite images. The use of sky cameras has shown to be a promising method for intrahour GHI forecasting [5, 6]. With the developments in deep learning, current machine learning (ML) models are able to extract information from such images and establish dependencies within sequential data. We propose a new hybrid deep learning model, composed by a convolutional neural network combined with long short-term memory (CNN-LSTM) layers, for intraday GHI forecasting. This paper mainly focuses on the 2-hour time horizon. However, different time horizons are also tested; 10, 20, 30, 60 and 180 minutes. This focus on intraday forecasting horizons is of interest for management of ADNs and microgrid energy storage systems and intraday electricity market trades.

A key innovative element of this paper is the use of images from publicly available webcams as well as combining images from different locations. Webcams typically present a side view of the sky, which brings cloud height information not present in all sky cameras and satellite images. The neural network is trained on a dataset of public webcam images

[1] A set of grid-tied controllable or partially-controllable energy conversion devices (e.g., solar panels, batteries and electric vehicles) that collectively provide ancillary services to the local and/or the bulk power grid.
[2] ADNs are distribution networks that have systems in place to control a combination of generators, loads and energy storage (collectively termed as distributed energy resources – DERs).
[3] A localized electricity network that can operate independently while still being connected to the main power grid, enabling it to generate, store, and distribute electricity as needed.

collected on EPFL campus and surrounding areas. Figure 1 shows two sample images taken at the same time from the webcams located on EPFL's campus. The two annotations, the green arrow and the blue rectangle, show the locations of two buildings on campus, building TCV and Rolex Learning Center. Figure 2 shows the location and the field of view (not limited by the red circle in terms of distance) of both cameras on EPFL's campus. The webcams used in this work are Roundshot's "livecam gen 3" webcams. The coordinates of camera "livecam EPFL BC" are 46.518391°N, 6.561931°E and those for camera "EPFL Place Maurice Cosandey" are 46.51776°N, 6.56662°E.

Fig. 1: Sample images from the webcams on the EPFL campus.

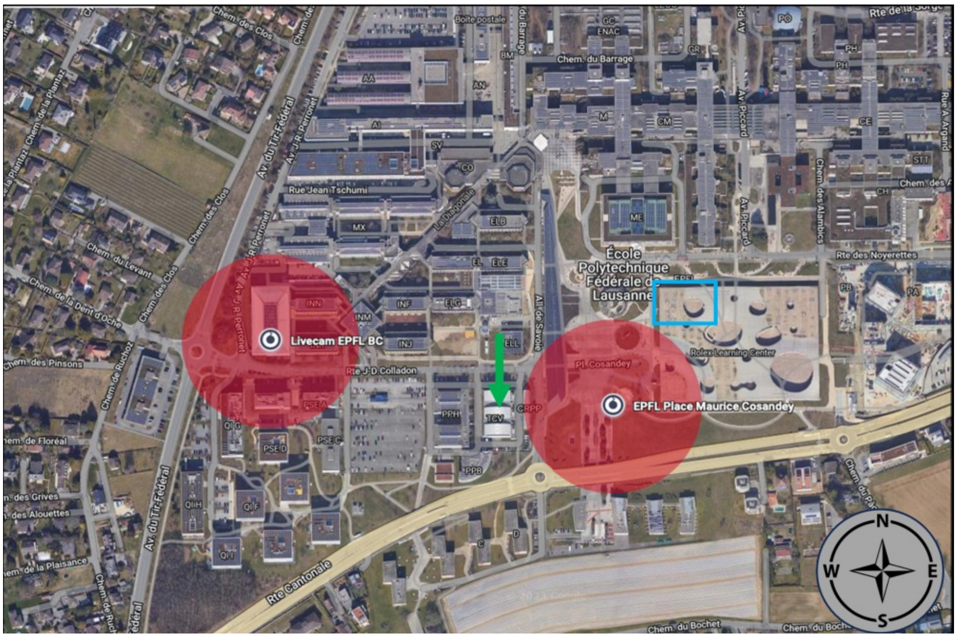

Fig. 2: Cameras' locations on the EPFL campus.

Table 1 shows a quick comparison between the livecam gen 3 and a "Basler" USB 1.8 megapixel camera (acA1300-200uc). The livecam gen 3 cameras were only used since they are preinstalled on campus. Using data fusion ML methods, multiple camera feeds are combined to train a neural network. Figure 3 shows the proposed deep neural network (DNN) architecture, which is described in more details in section 4. The methodology presented in this paper is scalable beyond EPFL because of the ever-increasing coverage of weather webcams.

Table 1: Cameras' Comparison

	Webcam	All-sky camera
Cost ($)	7400 - 8000	550 - 630
Spatial Resolution (pixels)	10752 × 2048	1280 x 1024
Temporal Resolution	1 panorama every 10 minutes	1 image every 1 minute
Field of View	panoramas up to 360°	up to 180° x 180°

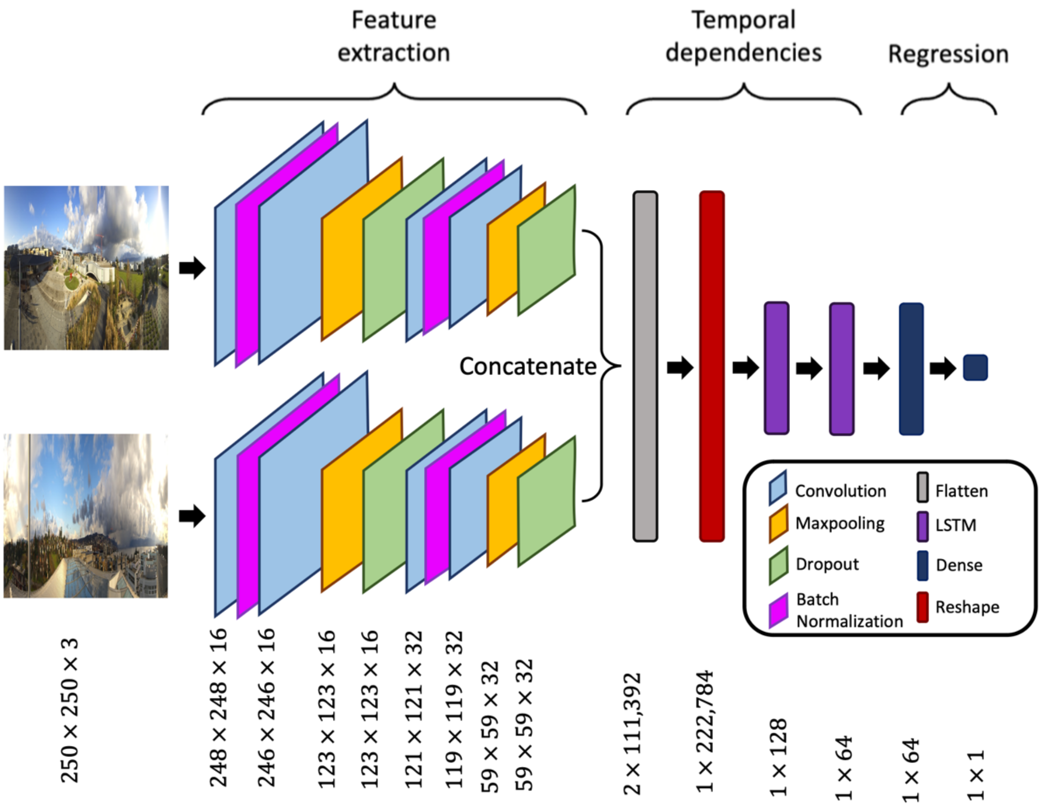

Fig. 3: Proposed CNN-LSTM network.

2. Data and methodology

This section presents the data collection process as well as the specifics related to the methodology of the CNN-LSTM network.

2.1 Data

The images used in this study are downloaded from Roundshot's website. They are 360◦ images with different resolutions. Two sample images are provided in Figure 1. Figure 2 shows the location of the two cameras on EPFL's campus. As previously mentioned, the main reason these cameras were used is due to the fact that they were preinstalled on campus. An alternative in case such cameras did not exist is to install outdoor 4K (3840 x 2160 pixels) security webcams that is a fraction of Roundshot's webcam price and provide images at a higher rate.

A python code downloads the images from the two different cameras from the public website, at a 10-minute frequency (as described on the webpage). All the timestamps of the downloaded images are used although some are missing for unknown reasons. Additionally, corrupted images were manually removed. Such images had distorted colors, i.e. very high contrast or very dark colors, or part of the image was missing. Only images taken between sunrise and sunset are used to make sure sunlight time is considered. As the images from both locations are different in width and height, they are scaled down using the "Pillow" python image library before being used as input to the neural network. Finally, an Apogee SP-230 all-seasons pyranometer collects local GHI measurements which are re-sampled at a 10-minute frequency to match the images.

Once all the images are collected, cleaned and scaled, we cross-check the timestamps at which the images are collected to match that of the pyranometer's measurements. The final dataset is then comprised of two images from two different cameras (collected at the same time "*t*") and a corresponding GHI measurement (representing the GHI at "*t* + 2 *hours*"). The data used in this study spans from end of December 2021 until the end of January 2023.

2.2 Methodology

The proposed CNN-LSTM network has two 3-dimensional input layers (as the images are RGB) followed by a time-distributed CNN block implemented to extract spatial features from each image separately. The CNN block is comprised of the following:

- A BatchNormalization layer between two Convolutional layers (Conv2D) with 16 filters and rectified linear (Relu) activation function
- A MaxPooling2D layer
- A Dropout layer with a dropout rate of 0.3
- A BatchNormalization layer between two Conv2D layers with 32 filters and Relu activation function
- A MaxPooling2D layer
- A Dropout layer with a dropout rate of 0.1.

The outputs of the CNN blocks are then concatenated together, flattened, reshaped and used as input to two consecutive LSTM layers. The two LSTM layers consist of 128 and 64 units respectively. Finally, a fully connected layers with 64 neurons is used leading to the output layer with a single neuron. The network is then trained with the scaled images and an early stopping criteria of 10 epochs is added. It should be noted that the '*restore best weight* ' parameter is set to '*True*' to make sure to restore the model weights from the epoch with the lowest error. It is worth mentioning that the hyperparameter tuning of the network was done via a grid search using the *sklearn* library in python. This was performed on the number of filters in the CNN block, the number of neurons in the LSTM and Dense layers as well as the rate of the Dropout layer. The architecture of the network is presented in Figure 3.

2.3 Evaluation metrics

The performance of the deep neural network is assessed using the following metrics: RMSE, normalized RMSE (nRMSE) and per unit RMSE (puRMSE) defined as:

$$RMSE = \sqrt{\sum_{t=1}^{N} \frac{\left(\widehat{GHI_t} - GHI_t\right)^2}{\mathrm{N}}}$$

$$puRMSE = \frac{RMSE}{GHI_{max}}$$

$$nRMSE = \frac{RMSE}{GHI_{mean}}$$

where N denotes the total number of testing timestamps, $\widehat{GHI_t}$ denotes the predicted value of GHI, GHI_{max} and GHI_{mean} denote the maximum and average value of GHI, respectively. We also compare the RMSE against the persistence model, in particular the smart persistence, which is treated as a benchmark. Both the simple and smart persistence models are commonly used benchmarks for irradiance forecasting. The simple persistence model is a very basic approach for predicting GHI. That assumes that the GHI will remain constant over time, i.e., GHIt+n = GHIt where n = 1, 2, ... is an index to the next time measurement. Usually, such model is used as benchmark for short-term predictions (<30 minutes). In contrast, the smart persistence (SP) model improves on the simple version by assuming that the relative GHI remains constant over the forecasting horizon. This is presented as such:

$$SP_{t+n} = \frac{GHI_t}{GHI_{t,cs}} GHI_{t+n,cs}$$

where the subscript *cs* represents clear sky conditions. It should be noted that the different models reported in Figure 4 outperform the smart persistent model except the CNN and ARIMA_{cam} models.

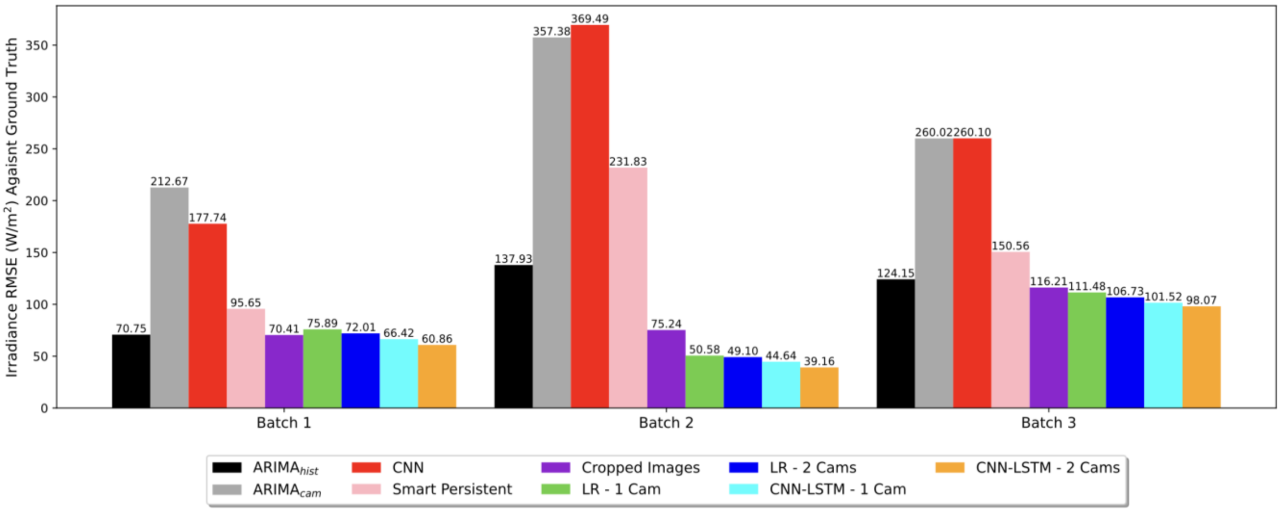

Fig. 4: Forecasts' RMSE comparison by model.

These metrics are used to compare different forecasting models such as an ARIMA time series model, a linear regression model and the proposed deep neural network with and without an LSTM layer. This research also compares the results of the proposed model, when using images from one camera only against adding the images from both cameras. Additionally, we test whether using the full images is better than using a cropped version that only captures the sky and clouds. Figure 5 illustrates a cropped version of Figure 1. Finally, the best model is benchmarked against Solcast's forecasts.

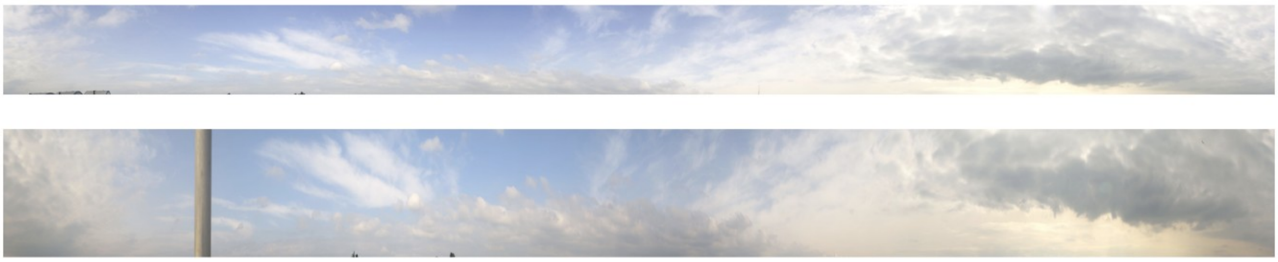
Fig. 5: Cropped sample images from both EPFL cameras.

3. Results and discussion

Figure 4 summarizes the aforementioned metrics for different models and data batches. As only one year worth of data is available, three different batches of images were from the full dataset to be used as out-of-sample testing samples. These batches cover a few days from different times of the year. Batch 1 includes images from January 27th, 2022, until February 10th, 2022, with four missing days due to the unavailability of data from the cameras. Batch 2 covers the data collected from July 7th to 19th 2022 with four missing days. Batch 3 covers eight consecutive days from September 18th to 25th 2022. As already mentioned before, once the data was cleaned and the images timestamps were cross-checked from the two cameras, some timestamps were lost. This results in having a different number of images for different days.

With regard to the time series approach, the ARIMA model was chosen independently for each batch by analyzing the autocorrelation and partial autocorrelation of the training data. The training data represents historical GHI measurements prior to the dates of the respective batches, i.e. for batch 1, the ARIMA model would be fit on historical GHI data from December 2021 until January 26, 2023. Additionally, for each batch, we test two different ARIMA models: one fitted on historical GHI collected at a 10-minute frequency (including GHI during the night) and one fitted on historical GHI collected only at timestamps where images are available. The first model is denoted as $ARIMA_{hist}$ and the second as $ARIMA_{cam}$. It should be noted that both models are seasonal ARIMAs. The fit was done using the "*Statsmodel*" library in python. This package provides the "*STLForecast*" function that decomposes the time series into a seasonal trend using locally estimated scatterplot smoothing (LOESS) to extract the seasonal, trend and low pass components.

Once the model is fitted, the autocorrelation and partial autocorrelation functions are used to identify the optimal lag order of the autoregressive and moving average components. These functions serve as a guide for selecting the dominant model. We further test and compare different models (i.e. different lag orders) using the Akaike information criterion (AIC) [7] and Bayesian information criterion (BIC) to choose the best model. The two measures provide a measure of goodness of fit for each model and by balancing between model complexity and model fit. The AIC value of a model is computed as AIC = 2k − 2 $\ln(\hat{\mathrm{L}})$ where k and $\hat{L}$ represent respectively the number of parameters and the maximized log-likelihood function of the model. The BIC is given by BIC = log(N) × k − 2 $\ln(\hat{\mathrm{L}})$ where *N* is the sample size. For batch 1, both ARIMA models had the same order of (5,1,3). In the case of batch 2, $ARIMA_{hist}$ had an order of (4,1,4) whereas $ARIMA_{cam}$ had an order of (6,1,4). Finally, for batch 3, both models shared the same order of (3,1,3).

As shown in Figure 4, both time series models perform worse than the proposed CNN-LSTM network for all three batches with a RMSE of up to 9 times in case of batch 2. The results were in line with our expectations as we know time series model perform poorly for intraday GHI predictions. An interesting result is the large difference in RMSE between the two ARIMA models in each batch as the $ARIMA_{hist}$ model, fitted on historical GHI with no missing timestamps, is able to better predict future GHI. It is also expected that this could be the case when it comes to images as having an uninterrupted sequence of images might improve the network's performance. To test the effect of the LSTM layer in the proposed architecture, the two LSTM layers are removed, and the new network (labeled CNN) is tested on the same three batches. The impact that the LSTM layers have on the predictions' accuracy is clear as the RMSE increases by a factor of up to 9.5 in the case of batch 2. This increase in RMSE was expected as the images have a temporal aspect that cannot be handled properly by the CNN layers
alone.

To test whether using images from two separate cameras provide a prediction improvement, we report the lowest error obtained by using images from one camera only. This is done for the linear regression as well as for the proposed DNN. In other words, given that there are two sets of images (one from each camera), we fit the linear regression and the DNN on each set alone and report the lowest error which is reported in

Figure 4. The models are labeled LR - 1 Cam and LR - 2 Cams for the linear regression with one and 2 cameras respectively, and CNN-LSTM - 1 Cam and CNN-LSTM - 2 Cams for the proposed network. The results of these tests show that including the two sets of images as input always improves the RMSE of the proposed DNN by more than 10% in the case of batch 2. Figure 5 shows the DNN's prediction and the ground truth of batch 2. From the previous experiments, we can conclude that including an LSTM layer along with two images yield the lowest RMSE error as depicted by model CNN-LSTM - 2 Cams in Figure 4. Additionally, a different model with "ConvLSTM" layers was tested. The RMSE was on par or worse than the proposed CNN-LSTM architecture. In batch 2 for example, the RMSE was around 10% higher. Additionally, it takes twice as long per epoch to train a network comprised of 3 ConvLSTM layers followed by two dense (fully connected) layers. The training time per epoch was 44 seconds compared to 19 seconds for our proposed network. It also requires more epochs for the ConvLSTM to converge compared to the CNN-LSTM.

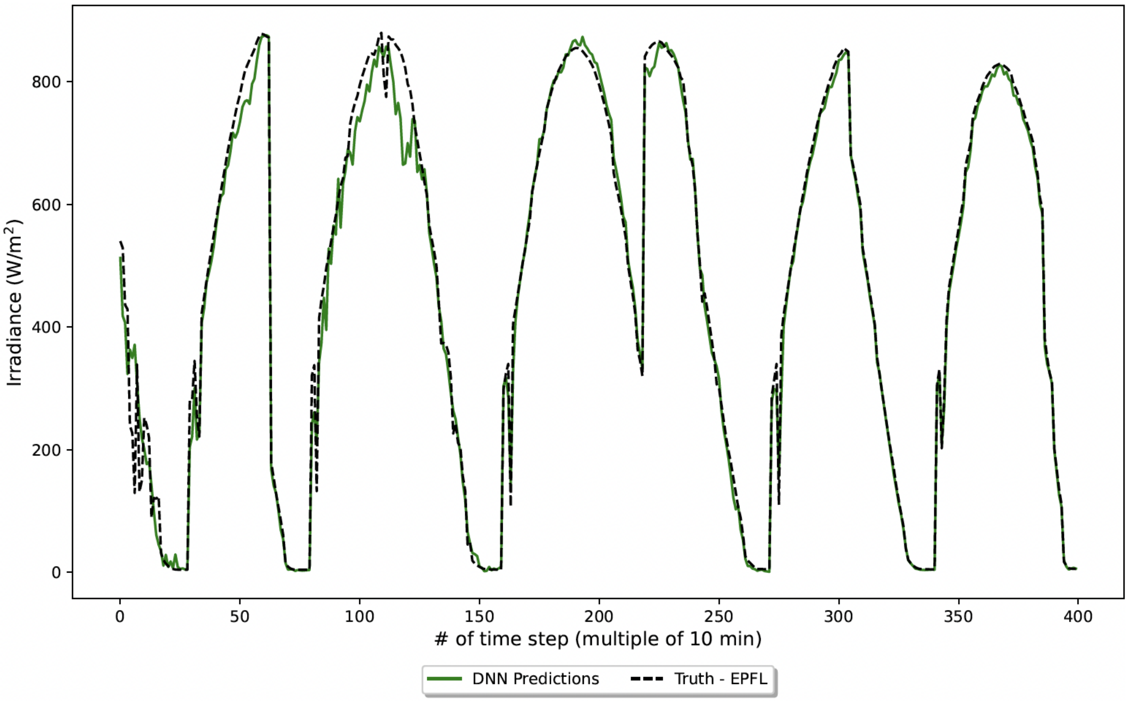

Fig. 6: DNN predictions (batch 2) and ground truth.

This research also explores whether using the full uncropped images (showing both the sky and surrounding areas) provides more information (and hence improved GHI prediction) compared to the cropped images. As expected, using the cropped images provides less accurate forecasts with a RMSE three times higher in case of batch 3. This is mainly due to the information that is being lost once the images are cropped. Such information could represent but is not limited to shadows and reflections. As the collected images could potentially show people, one use case of the cropped images is privacy protection as the image would be limited to the sky and clouds. This is specially the case for the first sample image in figure 1. To mitigate this problem without cropping the images, OpenCV's YOLO v3 (You Only Look Once) model for object detection was used. This is a pre-trained model that is optimized for object detection for 80 different categories. YOLO v3 was used to detect people in the raw images of the camera that might have privacy concerns. The detected people are then blurred with a gaussian blur from OpenCV. Finally, the proposed neural network is trained on the down sampled blurred images. The performance of the network is not affected for the three batches.

Additionally, using the same DNN architecture, we benchmark the accuracy of the forecasts against that of Solcast. For this, we first collect the forecasts of Solcast for the location of interest, EPFL (Lat: 46.5191, Long: 6.5668) in this case. Using Solcast's API, we can collect their GHI predictions 2 hours in advance at a 10-minute frequency which matches the frequency of the image collection. The data is collected for 12 days starting from March 28, 2023, till April 11, 2023 (March 30th and 31st are missing) while simultaneously downloading the images corresponding to those timestamps. This constitutes the testing dataset which includes 673 couples of images (673 images from each camera) and Solcast's predictions. Consequently, at every time step "*t*", two images and Solcast's forecast for "*t* + 2 *hours*" are collected. Once all the data was collected and the timestamps are cross-checked to match each other, the DNN is trained on all the data available (end of December 2021 till end of January 2023). The trained network is then used to predict the testing set. The RMSE between Solcast's prediction and the true measurements is computed as well as the RMSE of our predictions. The training process is repeated 10 times; in each iteration, the DNN is re-initialized with a new set of random weights. Figure 7 shows the daily RMSE of Solcast's predictions and the average (of the 10 runs) daily RMSE of the DNN's predictions.

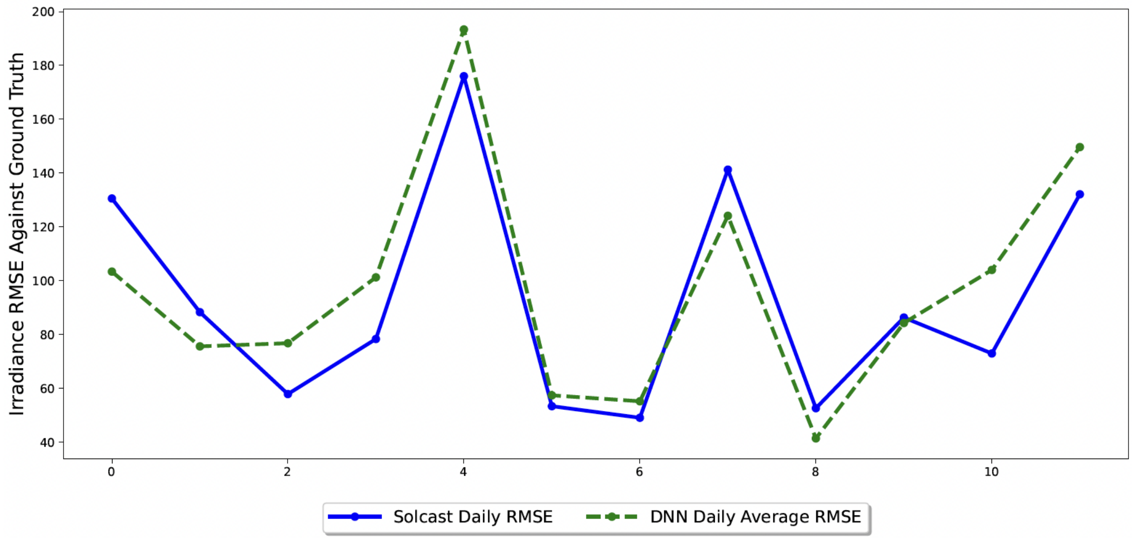

Fig. 7: Daily RMSE comparison between Solcast and the proposed CNN-LSTM.

From figure 7, we can clearly see that the GHI prediction of the proposed network based only on information from two webcams matches that of Solcast which uses satellite images and weather models. As noted on their website, their method for estimating GHI consists of multiple steps summarized as follows:

- Combining raw geostationary satellite images with atmospheric forecasts. This generates the "Solcast cloud model" that is the determinant
- Combining atmospheric pressure and water vapor, aerosols and ozone, and albedo (surface reflectivity) to form the "clear sky GHI".
- Combining the "cloud opacity" model with the "clear sky GHI" to predict GHI.

Our proposed method consists of simply using downsampled raw images from webcams without any additional information. The proposed DNN is also tested for different horizons. Results are shown in figure 8.

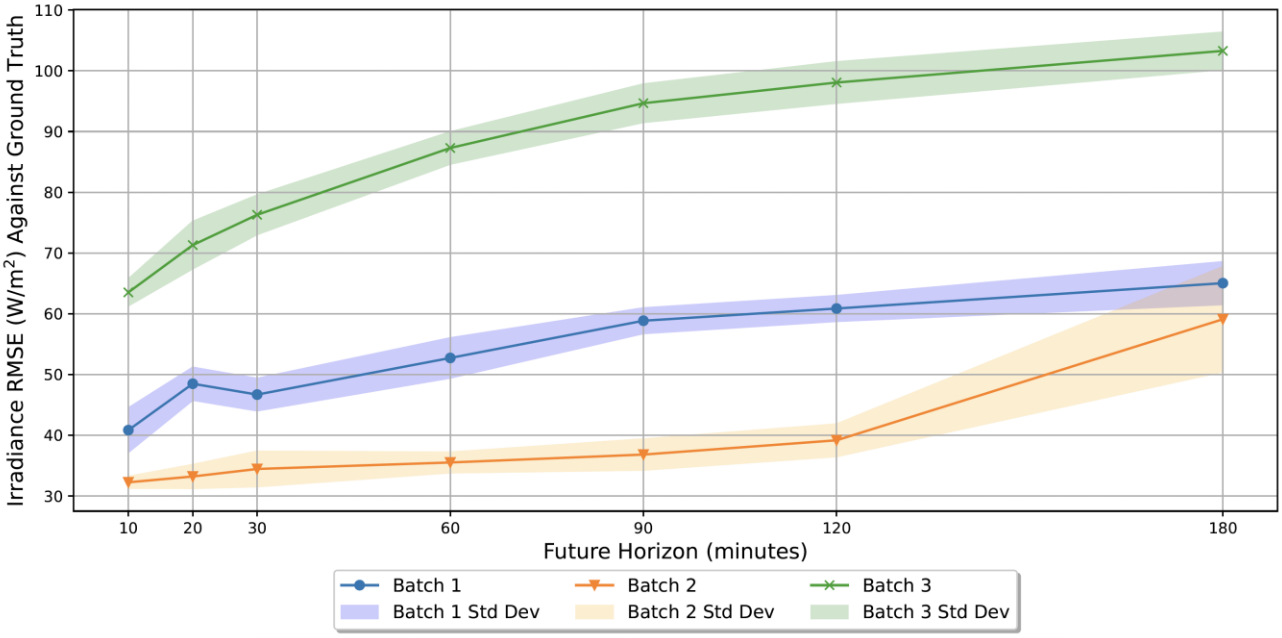

Fig. 8: RMSE by time horizon for the three testing sets.

The findings of this study are attributed to the characteristics of the webcam images. While these webcams may not capture the entire sky (as an all-sky camera would), especially at higher elevations, the wide field of view covered especially along the horizon allows them to capture cloud's movements across consecutive images. This can provide valuable information about cloud cover dynamics, which is the most significant factor in solar irradiance forecasting. This depth perception can help in identifying cloud layers and understanding atmospheric dynamics. For example, cumulus clouds are typically associated with fair weather, while nimbostratus clouds often indicate precipitation. The images, as depicted in figure 1, show distant clouds across a few kilometers. Additionally, webcams offer scalability, allowing to deploy multiple cameras to cover larger areas or multiple locations. This scalability makes them suitable for expanding monitoring networks or upgrading existing systems as needed. Roundshot already has a network of more than 400 active live webcams around Switzerland. There also exists many other public webcams that can queried.

Another important attribute are shadows cast by objects in the webcam's field of view that can serve as useful features for the network. The presence and movement of shadows across the images can indicate the position and intensity of sunlight, as well as the shape and orientation of objects. By analyzing shadow patterns over time, the CNN layers can learn to infer cloud cover, cloud shadows, and other atmospheric phenomena. These features can enhance the accuracy of GHI forecasts by providing additional context about the local weather conditions.

One potential limitation of this study is that the training process was done using local images and GHI measurement, i.e. both the cameras and the GHI measurements are located within a radius of 150-200 meters. A possible solution is to use estimates of GHI as reported by meteorological forecasting companies or data from nearby weather stations. With this in mind, transfer learning could be also a possible future direction. Transfer learning is a machine learning method where a model trained for a certain task (on a certain dataset) is used for a different but similar task. In our case, one could leverage this by training the proposed network with enough local images and GHI measurements and then using the pre-trained network with images from another locations to predict GHI measurements. Other directions that can be explored are adding more images, include meteorological data, or combine the existing network with a time series model. Additionally, a comparative analysis between the proposed webcam-based methodology and an all-sky camera shall be conducted. Such a comparison could offer valuable insights into the efficacy and potential advantages of our methodology in real-world solar irradiance forecasting applications.

4. Conclusion

In this paper, we present a method for intraday GHI forecasting focusing on the 2-hour time horizon. To the best of the authors' knowledge, we are the first to use images from public webcams instead of all-sky cameras. We also use the raw collected images with no pre-processing except for downsampling the images.

The research in this study shows that the proposed DNN outperforms other models (ARIMA, linear regression and CNN). The forecasts of the proposed network are compared with the predictions provided by Solcast - a global solar forecasting and solar irradiance data modeling company. It is shown that with information only from two public webcams, we are able to match their predictions. Additionally, this research highlights the importance of LSTM layers in handling time-series data even when it comes to images as removing it would drastically increases the RMSE. We attribute this result to the information content present in the side view images of the sky and clouds. We expect that by adding more webcams and by combining information from satellite and weather models, we will be able to further increase the GHI prediction accuracy.

The proposed methodology is very simple, computationally inexpensive and proves to perform very well to provide GHI forecasts for PV generation prediction. Improving GHI forecasting at the local level can be aggregated up to a regional or even national level and can help predict a country's imbalances in the transmission grid leading to optimal use of energy sources. With better intraday predictions, the dispatch plan (set a day in advance) could be followed more accurately and hence avoid imbalances that incur penalties and require use of non-renewable sources.

References

[1] Irenastat online data query tool (2023).

[2] F. Sossan, E. Namor, R. Cherkaoui, M. Paolone, Achieving the dispatch- ability of distribution feeders through prosumers data driven forecasting and model predictive control of electrochemical storage, IEEE Transac- tions on Industrial Electronics 7 (4) (2016) 1762–1777.

[3] J. H. Yi, R. Cherkaoui, M. Paolone, Dispatch-aware planning of energy storage systems in active distribution network, Electric Power Systems Research 190 (2021).

[4] R. H. Inman, H. T. C. Pedro, C. F. M. Coimbr, Solar forecasting meth- ods for renewable energy integration, Progress in Energy and Combus- tion Science 39 (6) (2013) 535–576.

[5] C. W. Chow, B. Urquhart, M. Lave, A. Dominguez, J. Kleissl, J. Shields, B. Washom, Intra-hour forecasting with a total sky imager at the uc san diego solar energy testbed, Solar Energy 85 (11) (2011) 2881–2893.

[6] J. Calbo, J. Sabburg, Feature extraction from whole-sky ground-based images for cloud-type recognition, Journal of Atmospheric and Oceanic Technology 25 (1) (2008) 3–14.

[7] P. J. Brockwell, R. A. Davis, Introduction to Time Series and Forecast- ing, 2nd Edition, Springer-Verlang, 2002.